PREDICTIVE HELICOPTER FLIGHT DATA MONITORING FOR FLIGHT SAFETY DESIGN

A Dissertation
Presented to
The Academic Faculty

By

Alek Gavrilovski

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Georgia Institute of Technology

Georgia Institute of Technology

May 2017

PREDICTIVE HELICOPTER FLIGHT DATA MONITORING FOR FLIGHT SAFETY DESIGN

Approved by:

Dr. Dimitri N. Mavris, Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Daniel P. Schrage
School of Aerospace Engineering
Georgia Institute of Technology

Dr. J.V.R. Prasad
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Kyle B. Collins
School of Aerospace Engineering
Georgia Institute of Technology

Charles 'Cliff' Johnson
R-ASIAS Technical Monitor
Federal Aviation Administration

Date Approved: January 12, 2017

No one trusts a model except the person who wrote it; everyone trusts an observation, except the person who made it.

Harlow Shapley

To my family.

ACKNOWLEDGEMENTS

My name is printed under the title of this work, but in practice there have been many influences, contributions, and help from a large and diverse group of colleagues, friends, and family that enabled me to finish this effort. Thanks to all my committee members. Dr. Collins helped with discussions and ideas, Mr. Johnson was invaluable in defining the direction of the work toward a practical implementation. Prof. Schrage had unique perspectives on the role of this work in a larger domain. Prof. Prasad had invaluable technical insights and challenges to this work. Prof. Mavris was ultimately the person who started me down this path and provided the opportunity to work on this problem. His support, guidance and understanding were enablers for this work to progress from idea to printed paper. Advanced Rotorcraft Technology, Inc. and in particular Jan Goericke and Joe English who provided their technical expertise and assisted in resolving issues with Flightlab®. I would also like to thank the research staff at the Aerospace Systems Design Laboratory who helped me develop the technical and other skills necessary to complete this effort. Much of my development and ultimate success is due to my interactions with my peers: Mike Abraham, James Arruda, Sylvester Ashok, Etienne Bouchard, Imon Chakraborty, Youngjun Choi, John Dykes, Sayan Ghosh, Ryan and Allison Jacobs, Justin Kizer, Mario Lee, Mark Lopez, Alexia Payan, David Rancourt, Mike Ward, and many others - thank you! Finally, there is my family, without whom none of this would have been possible.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

λ	nondimensional inflow
μ	advance ratio
ρ	Air density
σC_{d_0}	Profile drag coefficient
σ	rotor solidity
A	Disk area
C_D	Drag coefficient
C_L	Lift coefficient
C_P	Power coefficient
d_0	Breakpoint for dual-rate oleo
d_i	Compression distance of i_{th} landing gear strut
f	Flat plate drag area
FM	Figure of Merit
K	profile power correction
k	induced power correction factor
L_i	Load force in i_{th} landing gear strut
P	Power required
R	Rotor radius
T	Thrust
V_h	Horizontal flight velocity from flight data
v_i	induced inflow
$C_{i1,2}$	Damping constants for dual-rate oleo

$K_{i1,2}$	Spring constants for dual-rate oleo
ASIAS	Aviation Safety Information Analysis and Sharing
CAA	Civil Aviation Authority
CFR	Code of Federal Regulations
CVR	Cockpit Voice Recorder
DAP	Descent Abnormality Profile
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
EASA	European Aviation Safety Agency
FDC	Flight Data Clustering
FDM	Flight Data Monitoring
FDR	Flight Data Recorder
FOQA	Flight Operational Quality Assurance
GA	General Aviation
GPS	Global Positioning System
HFDM	Helicopter Flight Data Monitoring
HOMP	Helicopter Operations Monitoring Programme
HUMS	Health and Usage Monitoring System
ICAO	International Civil Aviation Organization
IHST	International Helicopter Safety Team
IMS	Inductive Monitoring System
KDD	Knowledge Discovery in Databases
LCS	Longest Common Subsequence
LOC	Loss Of Control
MKAD	Multiple Kernel Anomaly Detection
MLP	Multi-Layer Perceptron network
NTSB	National Transportation Safety Board
QAR	Quick Access Recorder

SHL	Single Hidden Layer network
SME	Subject Matter Expert
SMS	Safety Management System
SOP	Standard Operating Procedure
SPC	Statistical Process Control
SVM	Support Vector Machine
SysID	System Identification
VHM	Vibration Health Monitoring
VRS	Vortex Ring State

SUMMARY

Rotorcraft possess unique flight capabilities which make them well suited for a variety of challenging missions, including operations in hazardous conditions and under significant time constraints. Unfortunately, the safety record of helicopter operations is an order of magnitude worse than that of commercial airlines, which has prompted the rotorcraft industry to investigate strategies for safety improvement. Among the safety approaches taken in industry, Helicopter Flight Data Monitoring (HFDM) stands out as an effective means of identifying and mitigating emerging safety concerns before the trend results in an accident. Unlike traditional post-accident analyses of flight data, HFDM and similar monitoring systems operate by accessing, analyzing and acting on flight data recorded during regular flight operations. By comparing flight data to a set of condition indicators, known as safety events or exceedances, HFDM systems have demonstrated the ability to alert operators of unsafe occurrences, and provide objective data in support of mitigation strategies. However, when safety events are poorly defined or based on incomplete knowledge, the ability of HFDM systems to detect safety hazards is diminished, with reduced potential to help prevent avoidable accidents. As a consequence, the condition indicators used in HFDM require continuous refinement to achieve adequate detection performance. Additionally, the detection of any adverse condition can only be achieved if pertinent safety events have been defined beforehand. Mitigating this dependence on pre-defined notions of deteriorating safety has been the focus of several recent studies on the use of flight data for safety enhancements in aviation.

This work demonstrated the use of several modeling approaches, with different methodological components that enable a systematic analysis of operating conditions for the purpose of assessing the risk experienced at those conditions. The model-based approach is demonstrated with both static performance models and a nonlinear dynamic simulation. In the case of performance models, improved detection performance was shown compared to

existing safety events used in practice. These types of models may be readily employed in existing monitoring systems, at the expense of limited applicability within a narrow segment of the full flight envelope. The full methodology therefore makes use of the flight simulation software FLIGHTLAB, with the goal of establishing a process where the vehicle can be virtually flown to a critical state and the resulting information can be used in consequent monitoring of the hazard identified in simulation. The methodology is further supported by regression methods for implementation of the complicated simulation results in a monitoring system.

The present methodology complements typical HFDM systems through the use of physics-based modeling of the helicopter in flight. The main objective of the proposed approach is improved understanding of conditions that contribute to current accident rates through the use of physics-based helicopter modeling. By utilizing knowledge of the physical characteristics of the monitored rotorcraft in addition to the gathered data, a much richer set of derived information becomes available for analysis. Further, physics-based models may be used in a predictive fashion to identify safety events for a particular vehicle in simulation, prior to their use in an HFDM system. These benefits are demonstrated for several flight conditions known to contribute to negative safety trends in helicopter operations, with the overall goal being eventual improvements in the rotorcraft safety records through improved monitoring capabilities for these adverse types of operation.

CHAPTER 1

MOTIVATION

1.1 Motivation

The ability to hover and maneuver at low speed, carry external loads, and airlift victims of medical emergencies, have made helicopters an invaluable part of a modern transportation system. At the same time, the unorthodox operations and frequently challenging conditions in which helicopters operate have undoubtedly contributed to the perception of helicopters as more dangerous than other forms of aviation. While the helicopter accident rate can be compared to that of General Aviation (GA), the accident rate for helicopter operations is an order of magnitude more severe than that of commercial airlines [1] as shown in Figure 1.1.

Because of the unacceptable helicopter accident rate, there has been a growing interest in improving the rotorcraft safety record. In 2005, the International Helicopter Safety Team (IHST) was established [2] with the goal of reducing and eventually eliminating rotorcraft accidents. The IHST coordinates the efforts of national (Canadian, U.S., European) Heli-

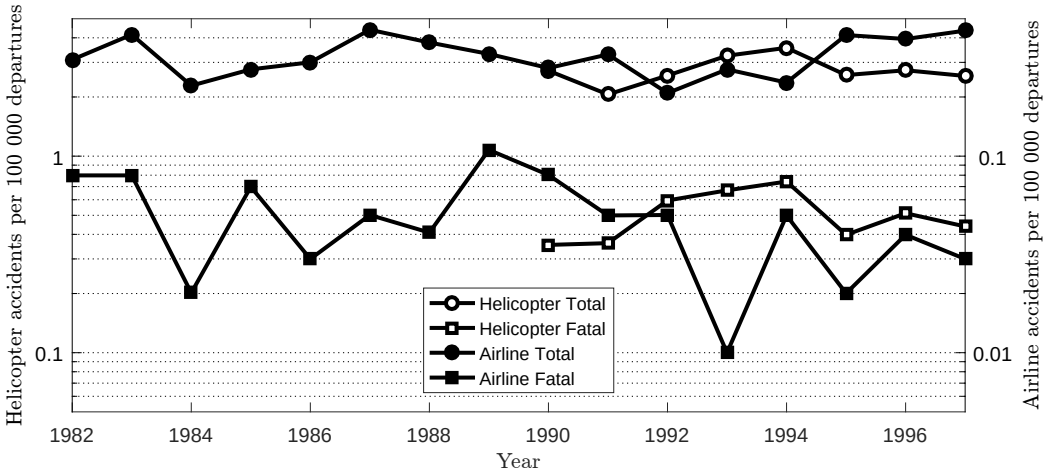


Figure 1.1: Helicopter accident rate compared to commercial airlines [1]

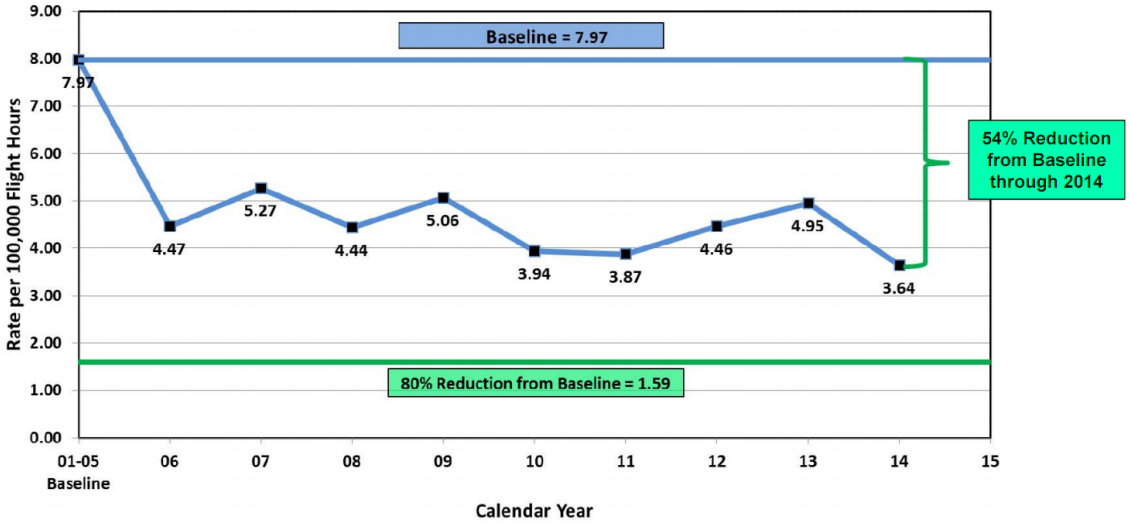


Figure 1.2: Rate of safety improvement for helicopter operations[1]

copter Safety Teams and works closely with industry, academia and government entities to achieve this goal. The high level of interest is confirmed by the *NTSB's Most Wanted* list of critical improvements to the transportation system, which has included improvements to helicopter safety for both 2014 and 2015 [3, 4]. The initial efforts of the IHST and similar organizations seem to have produced a significant decrease in accident rates, with the observation that following a period of continuous improvement, the rate at which helicopter safety is being improved has stagnated, as shown in Fig. 1.2. During the same period, the gap between helicopter accident rates and the results achieved by the commercial airlines has increased due to further improvements in commercial airline safety[5].

Figure 1.3 shows a categorization of helicopter accidents that occurred in the US during the years 2001, 2002 and 2006. It is evident that **pilot action** has been found to contribute to the majority of helicopter accidents. Another reason for the high proportion of accidents related to pilot error is the fact that over time, manufacturers have improved the reliability of their products to the point where mechanical failures are quite rare[6]. Therefore the overall number of accidents has dropped and reached a plateau where continuous improvement must be increasingly focused on human factors. An NTSB study of civil rotorcraft

[1] image from http://www.ihst.org/portals/54/2014%20US%20Accident%20Rate%20MEDIA.pdf

Table 1.1: Percentage of accidents related to pilot judgment

Source	% Pilot Action	% Data issues	Year(s)	No. of Accidents
Iseler & Mayo[1]	**70**	NA	1990-1996	756[2]
US JHSAT[8]	**84**	NA	2001,2002 & 2006	523
Canadian JHSAT[9]	**73**	52	2000	52
NTSB[10]	**69**	NA	2007-2009	26[2]
EHEST[11]	**70**	40	2000-2005	311
Harris,Franklin & Kasper[7]	**64**	NA	1963-1998	7618[3]

accidents between 1963 and 1998 found pilot error to be responsible for loss of control, collisions with objects/ground, and the direct cause for more than 50% of engine power loss accidents (through the improper use of fuel/air mixture)[7]. Over the years, several researchers surveyed large numbers of helicopter accidents with the goal of understanding causal factors and recommending solutions that can improve the stagnating helicopter safety trend [1, 8, 9, 10, 11].

In addition to pilot error, the studies agree that accident analyses tend to produce limited results when key pieces of information are missing or unavailable. The cause of the accident can be difficult to determine without a record of the actions performed by the pilots, which complicates investigations, especially when the physical evidence do not point to a mechanical failure. The studies conclude that **lack of information** is a major hurdle in performing accident analyses, and methods to obtain richer data from flight operations should be sought. Previous safety efforts in the fixed-wing domain concluded that objectivity through the use of data-driven methods is necessary for meaningful safety analysis and improvement in safety levels[12]. Because of the experiences in fixed wing safety, the use of objective data has been adopted as a basic tenet in IHST's activities.

Both the general lack of information, and the fact that most accidents are somehow related to the actions performed by the crew in the cockpit, can be addressed via on-board

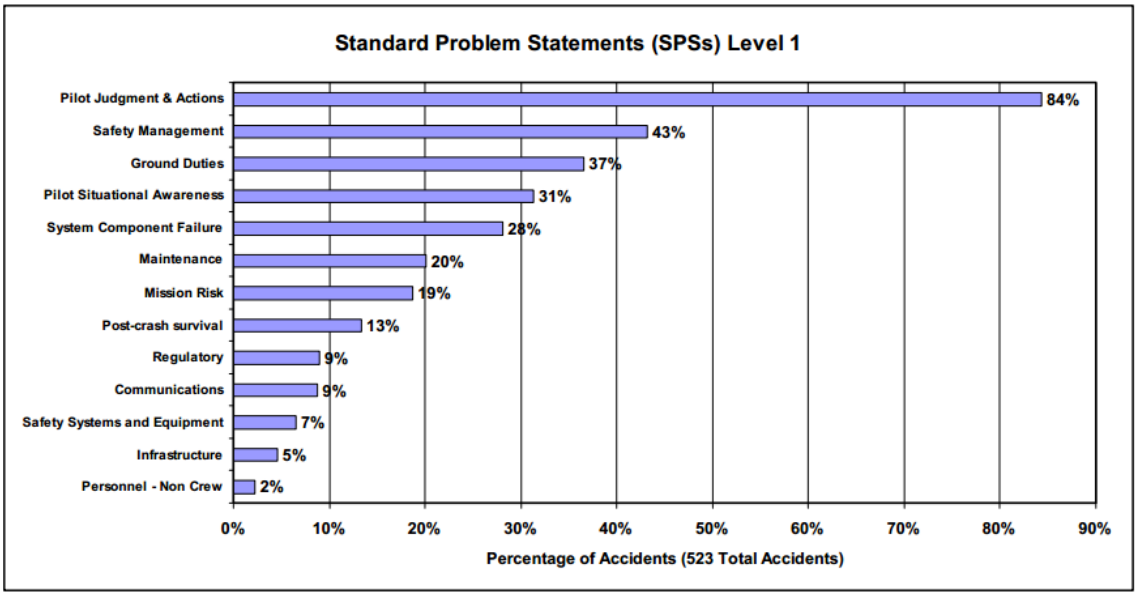


Figure 1.3: Pilot judgment is associated with the majority of accidents [8]

recording of flight data. Modern recording hardware has enabled much simpler download and analysis of data, so that daily analysis is possible, as opposed to only post-accident analyses using traditional flight data recorders. Thus in addition to accident investigations, the flight data can be used for daily evaluations of normal operations, with the potential of detecting negative trends before safety deteriorates toward an accident.

The systematic collection and analysis of flight data from routine operations for proactive safety improvements in aviation is known as **Flight Data Monitoring (FDM)** [13]. FDM systems have been successfully demonstrated by airlines over the past several decades [14], with measurable decreases in accident rates. Because of the safety benefits associated with FDM, rotorcraft operators have begun implementing their own versions of FDM, called Helicopter Flight Data Monitoring (HFDM).

Recently, a study by the European Helicopter Safety Team placed FDM technologies on its list of most promising technological means of achieving improvements in helicopter safety[15]. The NTSB Most Wanted List explicitly names FDM as a means to achieving

the needed reduction in rotorcraft accident rates. For all the above reasons, HFDM is becoming a more prominent component in safety systems employed by rotorcraft operators and regulators alike. Proper application of HFDM techniques holds the promise of proactive prevention of accidents and the potential to reduce fatalities in the helicopter industry. On the other hand, the set-up of an HFDM system requires extensive subject matter expertise and long periods of fine-tuning, which can result in decreased performance and thus reduced safety before the system is brought to full operational capability.

This section introduces the concept of HFDM through several representative sources of information and describes a generic version of an HFDM system which will provide context for further discussion.

1.2 Flight Data Monitoring

Operational data obtained from on-board measurements are important for a variety of activities from early development to operational service. During development, designers use flight data to assess aircraft performance and determine agreement between predicted and measured values. Toward the end of the design cycle, a thorough flight test program is conducted, in order to uncover the extent of the aircraft's flight envelope, identify any remaining deficiencies and provide data for certification purposes. In parallel, these flight data are used for system identification, where a model is created based on the relationship between control inputs, flight conditions, and vehicle responses. Such models are frequently used alongside pilot reports for handling qualities evaluations [16, 17, 18].

Flight data is of equal importance to the operators and regulators that are concerned with the safety of the aircraft after entering service. When flight data are not available, accident investigations have to rely on analysis of the crash site and physical evidence, witness reports and known facts about the aircraft/flight prior to the accident. As evidenced by

[2]NTSB reports accident numbers per vehicle category - light single engine vehicles are reported here

[3]Only light (single engine) helicopters are included here, but they account for over 90% of accidents in the period 1963-98.

Table 1.2: Increase in Flight Data Recorder capabilities [22]

Aircraft Type	Year entered into service	Number of parameters	FDR type	FDR data capacity
Boeing 707	1958	5	Analog	Mechanical limit of 5 parameters
Airbus 330	1993	280	Digital	128 wps[1] serial
Embraer 180	2004	774	Digital	256 wps[1] serial
Airbus A380	2007	>1000	Digital	1024 wps[1] serial
Boeing 787	2009	>1000	Digital	Ethernet

many cases throughout the history of aviation, these sources often lack enough information to uncover the reasons for an accident, especially when the accident is particularly severe or occurs in a remote area. In many cases, the only conclusive evidence in an accident investigation is derived from the readout of flight data and cockpit voice recorders. For this purpose, aviation authorities make the recording of on-board data using FDRs and CVRs a requirement on all airlines and in many other commercial operations[19, 20, 21]. Table 1.2 shows the progression of flight data recording on commercial airliners over a period of decades. As new aircraft enter service, the capabilities of the on-board recorders have grown in terms of number of parameters and quantity of data stored, as well as the sophistication of the storage medium. As capabilities of recording hardware have improved, so has the ease with which flight data can be accessed and downloaded for analysis. Quick Access Recorders (QARs) are a type of device that have contributed to this increase in data proliferation. QARs are similar in performance to FDRs, with the distinction of being mounted in an easily reachable location on the aircraft and having removable media that makes them ideal for frequent data downloads. Some of the more recent models can even transfer data wirelessly, further reducing the effort required to perform regular data analyses. This ability to frequently access and analyze data pertaining to regular flight operations

[1]Words per second, refers to the number of (usually) 4-bit data packets recorded each second. Multiple words can be combined to record parameters with greater resolution.

is one of the key enablers for widespread implementation of FDM systems[22]. By analyzing a much larger sample set, it is believed that a carefully designed FDM system can detect trends that indicate the potential for an accident to occur, but do so before the trend results in an actual accident. Thus the data-driven approach in FDM is focused primarily on *proactive safety* through timely response to any indications of eroding safely levels.

The UK Civil Aviation Authority (CAA) has published the CAP 739 Flight Data Monitoring guidance document, designed to help operators establish and utilize an FDM program. The document contains a wealth of information regarding the set-up and operation of FDM. In this document, the CAA defines FDM as **"the pro-active and non-punitive use of digital flight data from routine operations to improve aviation safety[13]"**. In the United States, FDM is known as FOQA - Flight Operational Quality Assurance, which the FAA defines FOQA as "**a voluntary safety program** that is designed to make commercial aviation safer by allowing commercial **airlines and pilots to share de-identified aggregate information** with the FAA so that the FAA can monitor national trends in aircraft operations and target its resources to **address operational risk issues (e.g., flight operations, air traffic control (ATC), airports)**[23]." The FAA further states that primary focus of FOQA is "to identify and reduce or eliminate safety risks, as well as minimize deviations from the regulations[23]."

FDM programs are referred to using a variety of terms, including Flight Data Analysis[24], Line Activity Monitoring Program[25], Flight Data Management[26], Military Flight Operational Quality Assurance[27, 28] and other names. Despite the different names, most of these programs operate under very similar principles, which are oriented toward proactive safety through timely detection of conditions that indicate the potential for an accident. Most FDM literature references the Heinrich Pyramid, shown in Figure 1.4, which is meant to convey the much higher frequency of low-consequence events relative to serious incidents and accidents. Thus a data-based safety approach such as FDM can improve the available safety information by detecting and tracking these events. Any safety initia-

tives can then utilize and act upon a much broader knowledge than what is available from accident analyses alone.

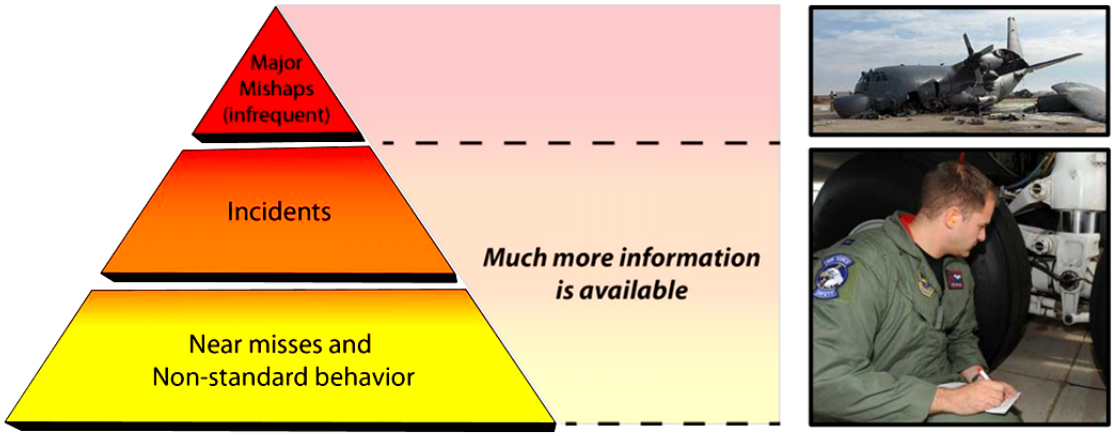


Figure 1.4: Heinrich pyramid of proactive safety[28]

Unlike the regulations pertaining to FDRs, national aviation authorities display a wider variety in their approach to FDM. The CAA requires commercial operators to maintain a functional FDM program if their aircraft meet certain criteria of size and passenger capacity. In the US, FOQA programs are voluntary, but can go through an approval process that helps ensure proper operation and affords certain legal protections. Overall there is less regulation surrounding FDM, but the acceptance level is high among airlines and is steadily growing among other types of operators.

To illustrate the operation of an FDM system, it is helpful to look at the flow of information through a representative FDM system as shown in Figure 1.5. The data are first imported into the analysis facility, either through physical movement of the storage medium or through wireless data transfer. These data are routinely analyzed by the downstream analysis system, and can sometimes be the subject of directed investigations, such as when a crew issues a request for feedback regarding some operational aspect.

Once the flight data are loaded onto the system, the flight recording is compared against a pre-defined set of conditions known as ”safety events”. In Figure 1.5 the analysis is performed within the box titled ”FDM analysis”, and a database of ”safety events” is usually

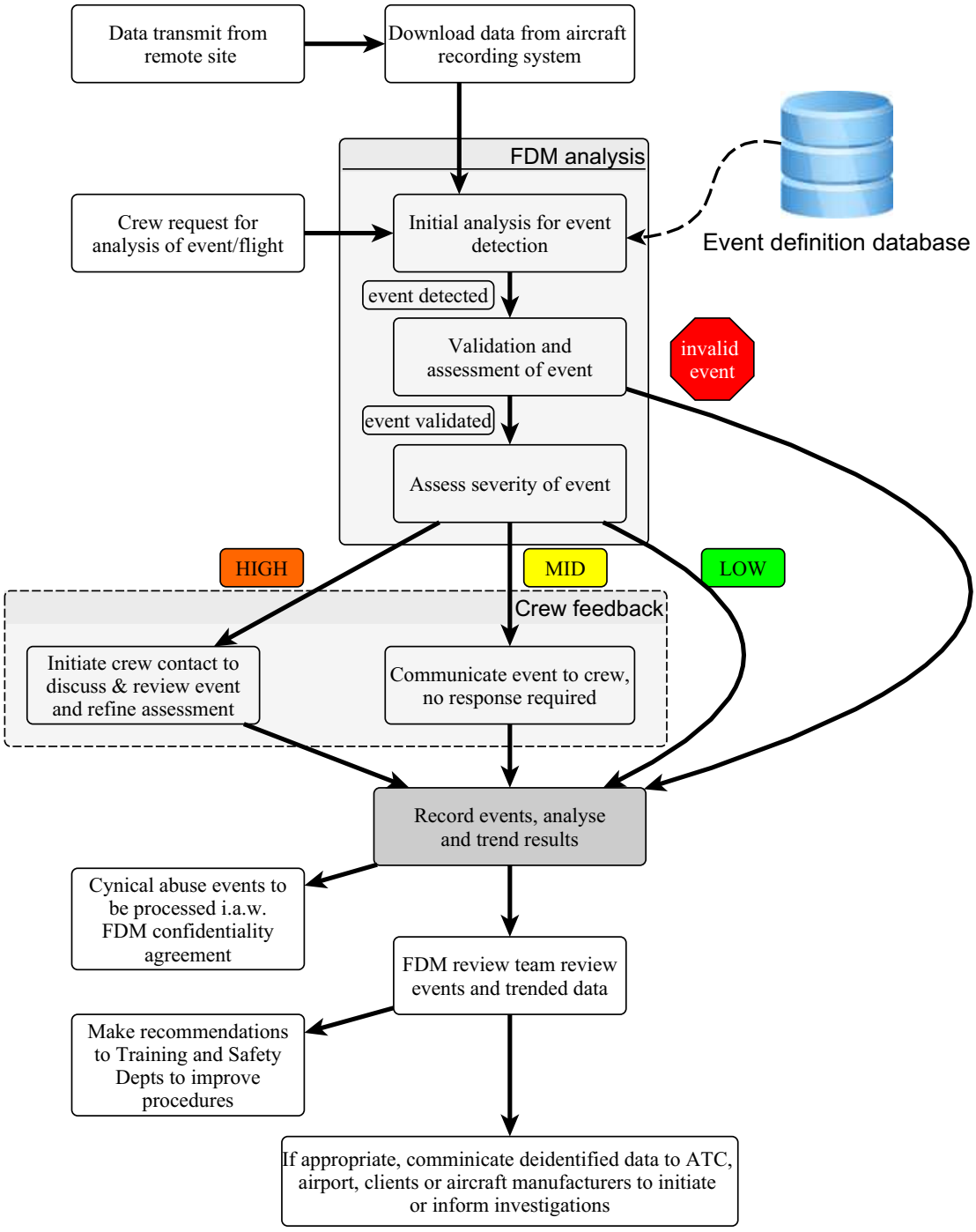


Figure 1.5: A typical diagram describing the flow of information in FDM

Table 1.3: Sample of basic operational safety events[13]

Event Group	**Description**
Flight Manual Speed Limits	Vmo exceedence Mmo exceedence Gear down speed exceedence Gear up/down selected speed exceedence
Flight Manual Altitude Limits	Exceedence of flap/slat altitude Exceedence of maximum operating altitude
High approach speed	Approach speed high within 90s of touchdown Approach speed high below 500ft AAL Approach speed high below 50ft AGL
Low approach speed	Approach speed low within 2 minutes of touchdown

maintained by the operator to enable the analysis. A safety event is a threshold or a set of thresholds on several parameters that is used to generate an alert when the parameters exceed acceptable levels, which would indicate that the level of safety has deteriorated[13]. For example, flight beyond the maximum allowable airspeed or exceeding a bank angle that is deemed excessive will result in an alert by the system. In addition to events, operators sometimes choose to record snapshot-type summaries of their operations, so conditions that are not necessarily related to safety are also tracked. An example would be the takeoff and landing times, or the indicated airspeed at the point when an aircraft passes through 10000ft, which could be used to demonstrate compliance with regulatory policies concerning noise procedures (CFR Title 14 91.117 [29]). CAA's CAP 739 [13] document and similar guidance publications frequently contain sample events, of which an excerpt is shown in Table 1.3. The full table also groups safety events according to phase of flight.

When setting up a new FDM program, a large portion of the effort goes to the definition

and fine-tuning of safety events. Subject matter experts are typically responsible for defining safety events before the FDM system can perform automated analyses, which makes the examples published in guidance documents especially useful to operators with new FDM programs. Safety events are usually created to correspond to the flight envelope, standard operating procedures, regulations and previous experience. The flight envelope is typically prescribed by the manufacturer of the vehicle[30], and is defined for each vehicle based on the manufacturer's analysis and experimentation during the vehicle's development. When operating the vehicle within the prescribed envelope, the level of safety should be reasonably high, so "safety events" are defined to correspond to cases when the envelope is breached[13, 23, 31]. The operator may choose to alter these limits to reflect their own SOPs, which are based on a combination of regulatory limits, the type of mission flown, and other operator-specific considerations. For example, an operator can limit the speed of an aircraft to less than the operating handbook value in order to extend the life of a particular component, i.e. for maintenance purposes[32]. Alternatively, if a limit specified in the pilot operating handbook is deemed too conservative, an operator might allow flight beyond the prescribed envelope. In this case, an FDM safety event would be set up to detect a parameter level that is already beyond the published envelope. For both situations, the FDM safety events reflect the operator's SOPs which are based on the flight envelope, but tailored by the operator to suit the level of risk acceptance, maintenance considerations, experience with the vehicle and the perception of conservatism already incorporated in the pilot handbook. Another reason safety events are defined is to prevent future occurrences of an unexpected/improbable accident that is encountered during operation. Operators keep adding new events as their experience with the FDM program grows, and operational occurrences point to potential dangers that can be monitored using flight data and analysis based on safety events.

Detected events pass through a validity check performed by a skilled analyst. The analyst uses their judgment and expertise to determine whether the safety event is a false

Severity

Catastrophic	5	5 Review	10 Unnacceptable	15 Unnacceptable	20 Unnacceptable	25 Unnacceptable
Hazardous	4	4 Acceptable	8 Review	12 Unnacceptable	16 Unnacceptable	20 Unnacceptable
Major	3	3 Acceptable	6 Review	9 Review	12 Unnacceptable	12 Unnacceptable
Minor	2	2 Acceptable	4 Acceptable	6 Review	8 Review	10 Unnacceptable
Negligible	1	1 Acceptable	2 Acceptable	3 Acceptable	4 Acceptable	5 Review
		Extremely improbable	Improbable	Remote	Occasional	Frequent
		1	2	3	4	5

Likelihood

Figure 1.6: Example Risk Tolerability Matrix [13]

alert or is indeed an unsafe occurrence. Valid events are then categorized according to their severity so that the downstream mitigation efforts can be prioritized and address the high risk safety deficiencies first. This assessment is performed using a combination of frequency of occurrence and the potential for damage/loss of life, which results in a risk matrix as shown in Figure 1.6.

In Figure 1.5, safety events of different risk levels are shown to feed into downstream portions of the FDM system where the operator decides on a course of action in response to the perceived risks. All safety events are recorded for long term analysis, but medium and high risk events usually necessitate more immediate reaction by the operator. This can be in the form of a bulletin or other communication, or direct engagement with the crew in cases where serious risks are identified or negligence is suspected[13, 23, 31, 33, 32]. FDM systems form the analysis portion of a larger Safety Management System (SMS) as outlined in ICAO's Safety Management Manual [34]. In the SMS there should exist safety support mechanisms to evaluate and manage risks, and a feedback channel that communicates findings back to the crews. Thus in Figure 1.5, the lower portion of the figure outlines some of the ways in which the results from the FDM analysis can be utilized, which is usu-

ally handled through organizational structure that is already in place with the operator[13], in the form of an SMS.

1.2.1 Results from FDM analyses

FDM analyses are usually done using software designed for the purpose of processing flight data [13], where the main capability of the software is the detection and reporting of instances where pre-defined *events* have been detected in the data, which consists of comparing incoming flight data against the event database. Most FDM systems go beyond mere detection and perform additional manipulations on the output of the event-based analyses, so that more informative results can be presented to the operator and a meaningful conclusion can be drawn regarding the safety/performance of the flight operations. A common example of such additional functionality are historical trending and statistical analyses performed using the rates at which events occur over a period of time, including time series representation, histograms, box plots and other statistical techniques. Another feature that is common across most FDM vendors is the ability to replay the flight using the position, attitude and accelerations from the flight data, along with any other useful information that can help create a detailed representation of the conditions that were experienced in flight.

Further analyses using FDM events designed to record ”snapshots” of typical operations are also performed, in order to present a high-level summary which the operator can use to evaluate the state of their operational performance and safety[31]. Monitoring such event-based summaries of their operations allows operators to objectively evaluate the adherence to SOPs, detect deviations before they grow to pose serious concern, and take measures to prevent accidents from happening in the future. These benign events can also serve as an indicator that a condition exists that should be tracked using safety events. For example, if an operator experiences a runway overrun, the history of touchdown speeds recorded through these ”snapshots” can be examined for a correlation between the landing speed and the incident. If a correlation exists, the operator can establish a safety event with different

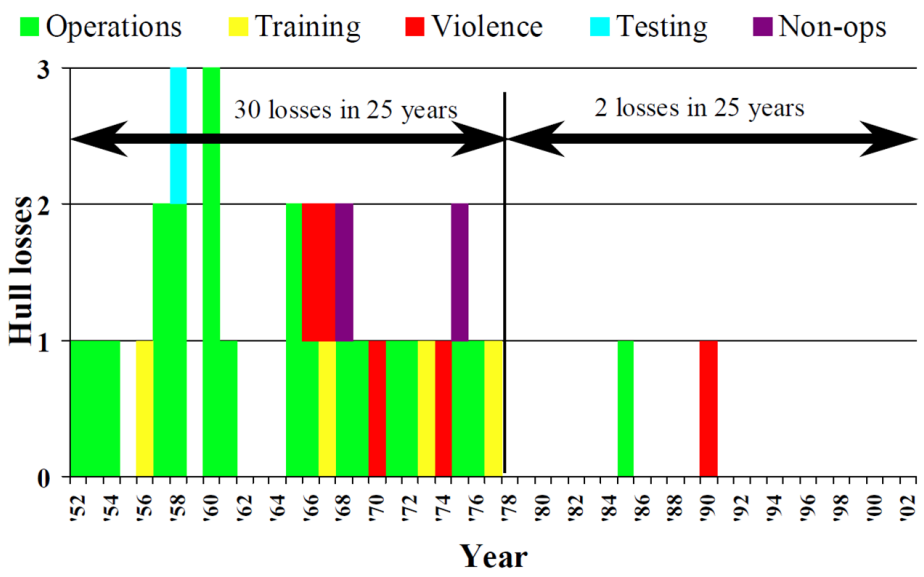


Figure 1.7: Reduction in hull losses following FDM introduction at British Airways[35]

severities corresponding to ranges of touchdown speed - thereby reducing the potential for future overruns. This is an example where typical aviation safety metrics, such as the *number of accidents per 100,000 flight hours*, are complemented by intermediate metrics in the form of FDM safety events, so that the operator can make safety corrections based on a much more detailed picture of their operations.

One of the earliest validations of the safety benefits achievable through FDM was the program implemented by British Airways during the latter portion of the 1970's[14]. Following the adoption of an FDM program, British Airways experienced a significant reduction of hull losses [35], as shown in Figure 1.7. It is results such as those demonstrated by British Airways that have prompted other airlines to set up their own FDM programs, which has contributed to the high level of safety experienced by modern airlines (Figure 1.8).

More recently, there has been increased interest in *data-sharing* due to the safety benefits of being able to perform analyses on large and varied sets of data from many operators. The FAA has helped set up a data sharing network called the Aviation Safety Information Analysis and Sharing (ASIAS) System[36]. This system allows regulators (the FAA)

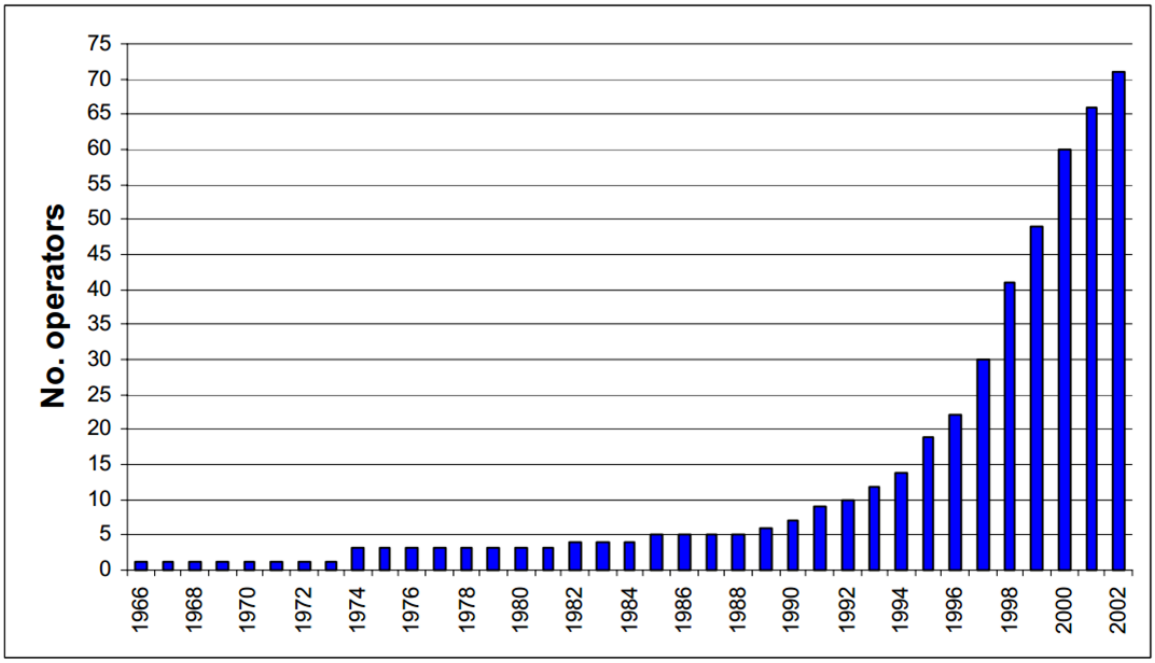


Figure 1.8: Rising number of operators with flight data monitoring[35]

and operators to analyze safety information from participating airlines, combine multiple and disparate sources of information (flight data from airlines, incident reports, weather and traffic data, etc.)[37] and to generate and disseminate safety information to individual operator and to the industry as a whole. The system is administered by MITRE, who performs the analyses on the aggregate data and generates guidance regarding common issues. For example, an operational issue detected within one operator can be examined across the entire base of participants, which has the potential to reveal the presence of similar issues among other operators much earlier than if the analyses had been performed in isolation[37]. For this reason, most US commercial carriers participate in the ASIAS program, and various efforts have been established to extend the benefits of ASIAS to General Aviation (GA) aircraft[38, 39], as well as to rotorcraft[33, 40].

FDM summary

The overview of a typical FDM system showed how the data flows from line operations (the aircraft) into the analysis system, where event-based analysis is performed, and showcased

some typical results and how they are used. The following observations can be made from the above reveiw:

- Modern commercial airliners have significant on-board flight data recording capability
- FDM systems analyze flight data from regular operations and have demonstrated safety improvements in airline operations. The analyzed set of data encompasses routine operations and can help detect negative safety trends before a catastrophic failure occurs.
- The primary analysis method used in FDM systems is ”safety event”-based detection of adverse safety conditions. Flight data are compared against a safety event database for any matches with known unsafe conditions. Benign ”snapshot” events are also used to provide a high-level summary of flight operations and for long-term trending.
- Creating a database of relevant safety events is a labor-intensive task, during which analysts translate system limitations, SOPs and general rules of flight into a set of algorithmic descriptions (safety event definitions) which can be used by the FDM system.
- Detected safety events are assessed and trended over time using traditional statistical techniques. Therefore it is important to have relevant events that have a meaningful correlation to the safety state of the system under observation.
- FDM is considered to be an essential aviation safety tool, with wide adoption and demonstrated safety benefits among airline operators. FDM regulation is specific to national aviation authorities, but the level of adoption generally exceeds the minimum regulatory standard.
- Data-sharing has been established primarily between commercial airlines, and efforts to include GA aircraft and rotorcraft are currently underway.

1.3 FDM in the helicopter domain

1.3.1 Health and Usage Monitoring Systems (HUMS)

Whereas manufacturers have improved the mechanical reliability of more modern helicopters[6], a significant portion of the increased safety can be attributed to the timely maintenance and early warning provided by Health and Usage Monitoring Systems (HUMS) [41]. These systems monitor the vibration levels and other signals coming from dynamic drive system components and alert operators to any unsafe increases in the monitored parameters that might signal deterioration of the component. The traditional approach to preventive maintenance is to assign schedules for replacement based on some initial testing, such as the number of hours that a component can be used before it needs to be inspected or replaced. By directly monitoring the state of the component, maintenance can be scheduled to coincide with actual deterioration of the part and eliminate premature replacement. Figure 1.9 shows the compromise required of traditional component life assessment, and the inefficiency when actual operations incur different life expenditure than the conditions used to assess a baseline component lifespan[42].

In addition to the cost benefits of maximizing part utilization, HUMS provides an added safety layer by alerting the operator to any premature failures, which has already been attributed to the safe conclusion of several potentially deadly incidents. In the 1980s the CAA performed an investigation of techniques to prevent catastrophic failures of dynamic components on helicopters operating in the North Sea's Oil and Gas industry, which resulted in quick adoption and widespread use of Health and Usage Monitoring Systems (HUMS) on rotorcraft. In 1999 HUMS/FDR systems were made mandatory on UK transport helicopters carrying more than 9 passengers[33]. The CAA refers to HUMS as Helicopter Vibration Health Monitoring (VHM) and has published extensive guidance on the topic[43].

HUMS has been researched and implemented in practice for several decades, and many of the techniques have been in use even longer, since much work had previously been done

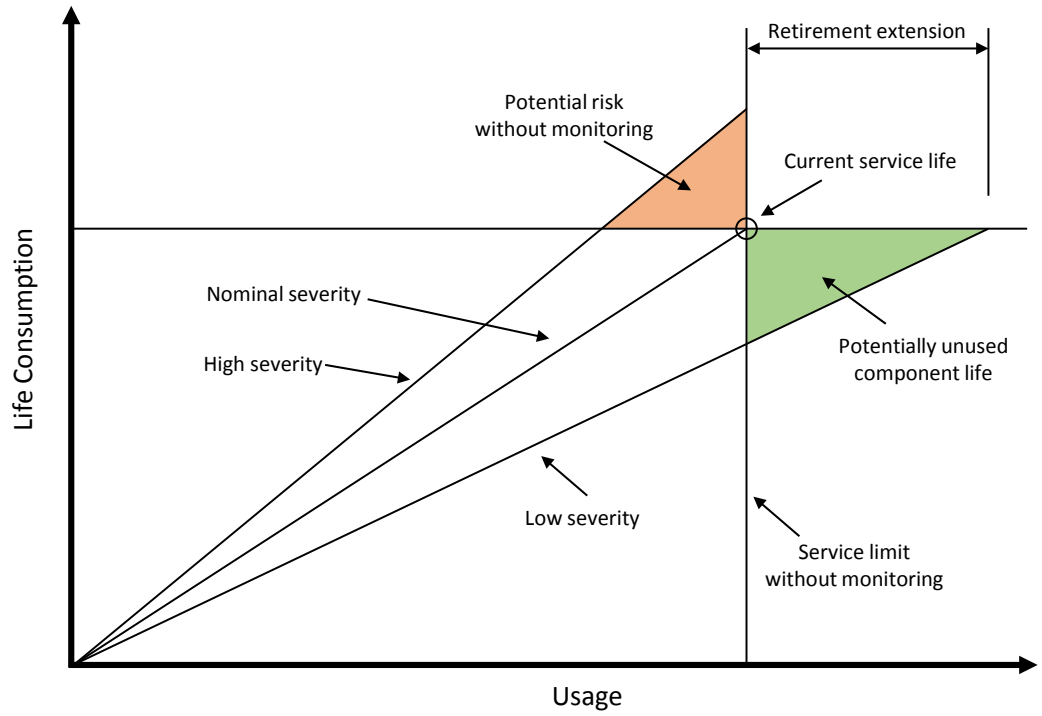


Figure 1.9: Traditional vs. HUMS based part life assessment[42]

for nuclear plant and rocket engine monitoring[44]. One of the fundamental HUMS techniques in practical use is Statistical Process Control (SPC). Once a system is in operation, SPC establishes nominal levels of variance in the monitored signals and generates and alert when the levels are exceeded. This gives rise to the term *exceedance*, which is sometimes also used to describe safety events in the context of FDM. The principle is very similar for both, in that they are condition indicators, with the primary difference being the type of parameters tracked and the level of expert judgment needed to establish baseline values for the safety event thresholds. In HUMS, most of the focus is on tracking the deterioration of dynamic components such as the rotor [42], rotating shafts, engines[45], gears and bearings[46]. For such components there are frequently well-specified limits on allowable vibration or noise spectrum, or the limits can be specified during the initial phase of system operation, and selected in a way that balances false detection and missed alarms[47]. For components that aren't directly monitored using sensors, regime recognition is used to detect the current operational state, and correlate the damage incurred in flight based on *a*

priori experimental evaluation of the damage profile for each flight regime[48].

Most HUMS systems, however, are only available on large and advanced helicopters that have digital engine controls, or otherwise have the on-board capability to monitor and record data from sensors mounted to the dynamic components. As such, HUMS typically serve the high-end of the rotorcraft industry. However, Iseler and De Maio[1] noted that large helicopters involved in commercial operations have a relatively lower accident rate, much closer to that of commercial airlines. Therefore any safety improvement effort must consider the light helicopters operated by small operators.

1.3.2 Helicopter Flight Data Monitoring

The Helicopter Flight Data Monitoring Toolkit published by the IHST [49] defines HFDM as **"A systematic method of accessing, analyzing and acting upon information obtained from flight data to identify and address operational risks before they can lead to incidents and accidents."** Compared to fixed wing FDM, HFDM systems have to address rotorcraft-specific considerations in order to be an effective safety tool, which is accomplished primarily through re-defining safety events to reflect the limits of safe helicopter flight[13, 50]. The helicopter *flight envelope*, the region in which safe flight can be expected, tends to be quite complex and exhibit variations with flight and environmental conditions, which makes it more difficult for pilots to maintain safe flight. A common definition of *Loss of Control* states that it occurs "because the aircraft enters a flight regime which is outside its normal envelope, usually, but not always at a high rate, thereby introducing an element of surprise for the flight crew involved[51]." For example, helicopters can be subject to hazards even while stationary (with rotor turning), if the pilots inadvertently move the main rotor controls to an extreme position, which could result in excessive rotor flapping and a rotor blade strike. The same control input in a fixed-wing aircraft that is stationary is fairly inconsequential. For the operator, regulator, or any other entity concerned with monitoring the safety of helicopter operation through HFDM, this com-

plexity translates into added difficulty when defining the safety events which are used by an HFDM system. This difficulty was reflected in the experience of early HFDM investigators. In particular, the CAA cooperated with North Sea operators to understand the challenges surrounding FDM use on helicopters. These studies are described below.

1.3.3 The Helicopter Operations Monitoring Programme (HOMP) Trials

Around the same time that HUMS became required in the North Sea, the CAA and Shell Aircraft sponsored a project to investigate the feasibility of using routine flight data for safety purposes, in addition to maintenance scheduling and accident investigation. This trial is known as the Helicopter Operations Monitoring Program (HOMP)[33]. The initial study involved a single AS 332 Super Puma helicopter used for data collection over a period of one year, and leveraged techniques developed by British Airways during their own early work on FDM. The HOMP trial demonstrated the extraction of meaningful safety information from data gathered on a flexible aircraft platform such as the helicopter, and provided mitigation strategies to issues detected through the use of HFDM. Figure 1.10 shows the basic concept of HOMP.

A second study was later performed [32] which evaluated the challenges and benefits of applying HFDM to different operators and helicopter types. The analysis was extended to a second operator of the same helicopter type (Super Puma), and the study revealed common operational risks which were recognized by both operators but received different treatment due to the operators' approach to risk. A second type of helicopter, the Sikorsky S76, was also added and evaluated. Initially, the S76 did not have GPS (Global Positioning System) data available which hindered the HOMP analysis. The importance of these data were quickly recognized and the requisite recording capability was installed[32].

Unlike HUMS, the safety events used in HFDM are based on more varied sources of operating limits (pilot handbook, SOPs, airmanship and experience) and the SPC approach to establishing nominal parameter values is less effective at creating valid safety event

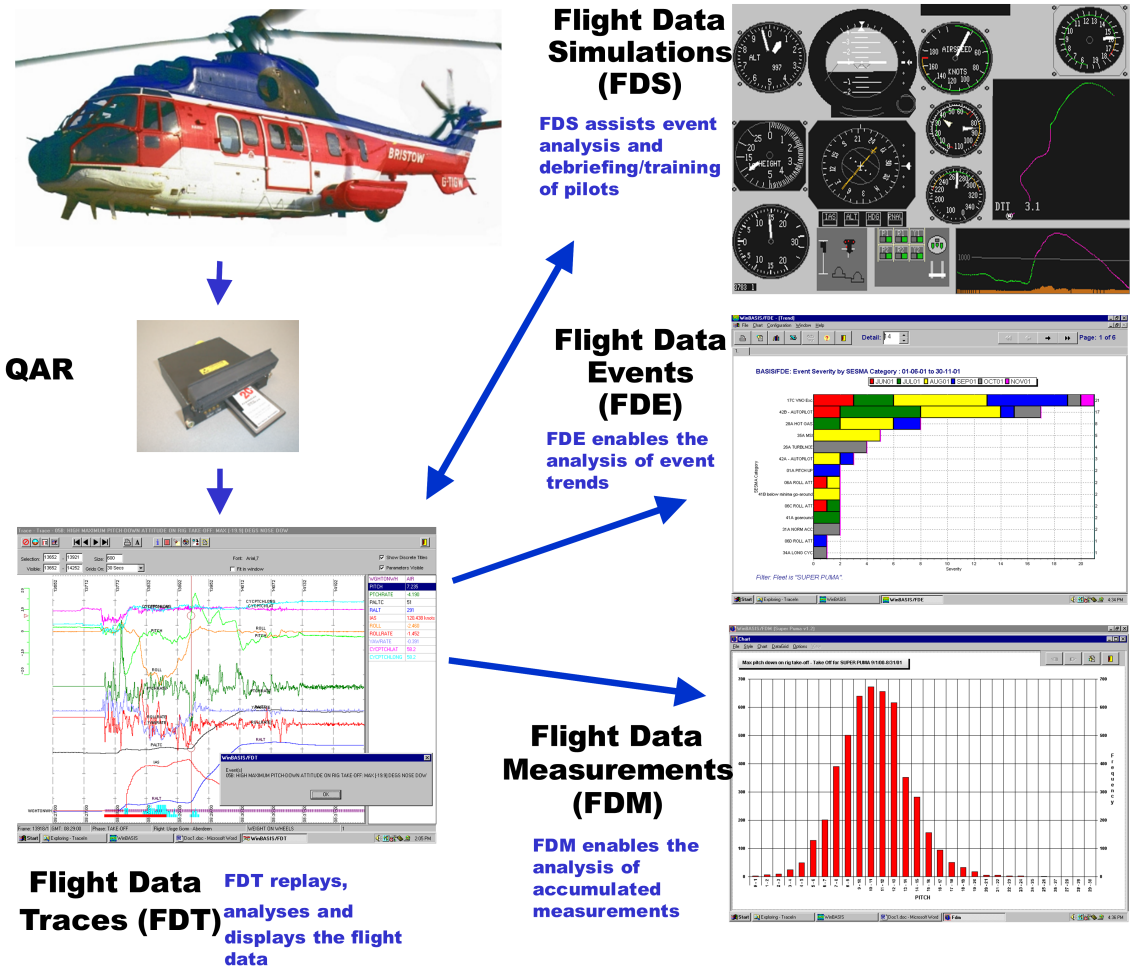


Figure 1.10: The HOMP system[32]

limits, particularly because of the wide range of values the parameters could exhibit while staying within safe boundaries of flight. The HOMP trials showed that following the initial set-up of a safety event database, continuous fine tuning is required until satisfactory levels of system performance are achieved. This performance is usually measured by the analysis team in terms of the number of false detections and missed alerts. False alerts occur when a safety event is triggered, and the detection is erroneous or the analyst deems the operation to be safe. Because analysts validate safety events through inspection of triggered alerts, such false detection presents a nuisance and increased workload, which in turn reduces the performance of the HFDM system. Because the focus is on safety, false alerts are preferred

to missed detection, and the parameter levels that make up safety event definitions are set to minimize or eliminate missed detection at the expense of increased numbers of false alerts. During the HOMP trial, 21 out of a total of 85 event definitions were reconfigured to reduce the number of false alarms or help the detection of certain events. The original rate for false alarms was around 99.95%, down to 90% toward the end of the program through the use of bad data detection and correction processes [33]. Considering that not all flights produced safety event triggers, the 90% rate equates to about 2-3 false alarms per flight. False events were removed manually through inspection, and later on employing logical tests when possible (multiple events strung together if occurring in quick succession). The researchers recommended that further work should be done to achieve a balance between false/nuisance alerts and detection performance [33, 32].

Some of the benefits reported by the researchers involved in the HOMP trials are immediate reductions in the incidence of particular events. One example given by the investigators cited the observation that helicopter crews flying the Sikorsky S76 were operating at high speed with the landing gear extended. The gear doors on the S76 have a structural limitation that could be exceeded at high speeds, so care must be taken to either retract the gear or limit the forward speed. After a number of safety events of the ”high-speed with gear down” type were detected, the operator communicated the importance of avoiding exceeding the gear limits to the crews. In the period immediately following this communication, the number of such events dropped for a period of time, before detection rates of this particular event type began creeping up. This experience demonstrated the effectiveness of having an objective view of the operations, as well as the need to monitor and communicate findings continuously for best results.

Conclusions from the HOMP trials were incorporated into data monitoring guidance materials, particularly in later revisions of the CAP 739 document[13]. The HOMP trials' experience with the definition of helicopter-specific safety events resulted in a *list of example safety events* that can be used by new HFDM programs to populate their safety

event database and provide the initial functionality of the HFDM system. These example events can be found in the appendix of the HOMP documents and CAP 739[13, 33, 32]. It is understood that operators and the analysts charged with the task of defining and maintaining events would modify the sample events in a way that reflects the specifics of their own operation and desired safety level, as well as define any additional events that may be required.

The two HOMP trials were limited to just two helicopter types that had on-board data recording capability already installed in the form of HUMS. In the case of the S76, additional GPS recording capability was added, but a general analysis of various on-board recording hardware options was beyond the scope of the study. In practice, the differences between helicopters and their on-board recording capabilities need to be understood before the lessons learned in one program can be applied to the next.

1.3.4 The Small Helicopter Operational Monitoring Programme (Small HOMP)

In 2010, the European Aviation Safety Agency (EASA) reported on a project that evaluated HFDM intended specifically for operators of small fleets of light helicopters[52]. The study had a mandate to "evaluate the potential safety benefit of applying HOMP to light helicopter operations and then to recommend a suitable FDM specification". To accomplish this task, the EASA researchers cooperated with helicopter operators that performed training, VIP transport, aerial work and personal flying mission types.

Part of the study was aimed at understanding the hardware requirements for HFDM on helicopters that otherwise do not have a data recording capability. Light helicopters present unique challenges for HFDM as a direct result of their small size. The weight penalty and packaging constraints mean that recording hardware must be *light and small*. The recorders that are suitable for applications on airliners and on heavy helicopters, may be exceedingly heavy for application in light helicopters, and may provide limited utility if the helicopter does not already have the requisite on-board sensors. In addition to weight,

Table 1.4: Small HOMP parameter set recommendation[52]

Set	Parameters
Minimal	GMT date and time; GPS latitude, longitude, elevation, ground and vertical speed; heading and yaw rate; roll attitude and rate; pitch attitude and rate; X Y Z accelerations
Recommended	Minimal + Engine temperature, torque; Oil temperature and pressure; Rotor RPM; Air Data (airspeed, OAT, vertical speed);WOW, Fuel level and rate; particle detection
Desirable	Recommended + Radar altimeter

many of the sensors and hardware available on larger helicopters are not cost-effective on smaller and less expensive helicopters. The EASA team reviewed a number of light data recorders for small helicopters, that either interface with on-board data sources or function in a standalone fashion to provide data for HFDM analysis. The report [52] contains a detailed comparison of the recorders available on the market at the time of writing, including an evaluation of the parameters available on each recorder device. Another useful source of information on helicopter flight data recorders that are suitable for HFDM, as well as software and HFDM services, can be found on the Global HFDM Steering Group website where a list of vendors is kept and periodically updated[53]. Based on the surveyed hardware and the flight parameters available across those devices, the researchers developed three sets of parameters. These parameter sets, in order of increasing utility to HFDM programs are shown in Table 1.4.

The Small HOMP study used the sample events defined during the original HOMP trials as a starting point, and modified them for the different missions and helicopters used. The report notes that due to the absence of certain parameters, not all of the original safety events could be implemented, but that the remaining safety events were useful in creating an effective HFDM system. Again, safety events were fine-tuned and continuously adjusted throughout the length of the study to achieve the best balance between detection performance and false alerts. In contrast to airlines and other large operators, the Small

HOMP trial found that the *limited engineering resources* available to a small operator can reduce the amount of fine-tuning possible, and affect the performance of the overall system for a longer period of time while the definitions of safety events are optimized. Assigning severities to each event and periodic review and evaluation of the results are tasks that also require significant human involvement. The Small HOMP report contains a detailed description of all the safety events developed during the project, coupled with an excellent analysis of the safety context for each event, and examples of how HFDM is applied to missions with different safety requirements[52].

Overall, the Small HOMP study produced positive results, especially considering the increased difficulty faced by smaller operators. A list of benefits is given in the report, primarily pertaining to training, safety, and maintenance improvements, but also for scheduling and leasing/contracts. The positive results are especially relevant in light of recent developments that promise to increase the number of small helicopters equipped with recording equipment of some sort. Airbus Helicopters (previously Eurocopter) pledged to install recording hardware as standard on all its products, beginning in the year 2015[54].

1.4 Observations

1.4.1 Challenges

Some issues faced by FDM programs are cultural, such as the guarded acceptance of FDM in the helicopter arena. There are even reports of operators that use safety as competitive market advantage, but fortunately this trend is subsiding[55]. Other problems arise due to the issue of implementation cost and fears of regulatory oversight. As researchers interested in the technical/implementation aspects of HFDM systems, we cannot directly affect cultural or cost issues. However, by focusing on improving the capability of HFDM systems we hope to generate a positive interference loop where increased performance leads to improved safety and a better justification for the costs involved, ultimately resulting in wider acceptance of HFDM systems. Below are some of the specific issues that limit the

capabilities of current HFDM systems in practice.

- *A priori* safety event definitions

  The vast majority of FDM systems operate by comparing flight data against a set of pre-defined conditions known as "safety events". If a condition has not been defined and is present in the system's database, then the operator is not protected against such occurrences. For this purpose, expert input is required early on to establish a usable database of safety events that can provide basic functionality that can be extended over time.

- Variability in the level of data content and quality

  Some data sets contain only vehicle position, velocity, while other more advanced recorders collect everything from pilot input, to air data to engine performance parameters. There are regulations that specify minimum parameters and therefore the lowest acceptable level of content in the data, but these types of regulations do not apply to most helicopter operators and especially to small operators. At the same time, committed operators frequently surpass those requirements, recording many more parameters than actually required. This variability in data content presents a difficulty in developing a HFDM methodology that performs with the same level of detail and accuracy for the operators with incomplete data, as it does for the operators with rich data. Because the content of the data is directly related to the recorders used, the type of helicopter and available sensors, and the willingness of the operators to invest in more advanced on-board data acquisition, the typical HFDM system is dependent on and constrained by the quality of the incoming data. HFDM systems have to be adapted and make the best use of the available data without the ability to affect the quality of on-board recording, at least not directly or immediately.

- Lack of traceability in event definition process

  Safety event definitions come from several sources, including the manufacturer (flight

manual), regulators (speed, altitude and airspace limits), the operator (SOPs), and good airmanship. However, most HFDM programs begin by setting operational limits that stem from the flight manual supplied by the manufacturer of the vehicle in question.

The manufacturer envelope is created before a vehicle enters operations, and is based on a subset of the conditions that the vehicle can encounter in real life. The flight test programs that are conducted to explore the flight envelope and create the guidance that is found in flight manuals seek to be as exhaustive as possible, but they simply cannot explore every possible condition in which the vehicle might be used. One reason for this limitation is that the flight test program has a finite budget and has limited time to perform the investigation.

- Unobserved safety deficiencies

  More importantly, some conditions are so unusual and seem so improbable that a flight test program fails to identify them as parts of the flight envelope to be explored. Hence there is a potential for dangerous conditions to remain unidentified until they are encountered by the operators. When such conditions lead to a notable incident or accident, guidance is implemented in the form of airworthiness directives (AD), and in the case of HFDM, new safety events that track the incidents of such conditions in the future are defined. This means that while HFDM and other data monitoring systems seek to proactively prevent future incidents, they have an inherently retrospective nature.

- Continuous fine tuning of safety events

  Event limits are frequently updated throughout the program as more knowledge becomes available about the conditions they are intended to detect. This need for constant rework limits the usefulness of past results as well as requires significant resources to perform the changes, which can be a detriment for potential HFDM

operators, especially smaller operators with limited human resources.

- Retrospective nature of FDM

  One of the main shortcomings of the current practice is the inherently retrospective nature of most FDM systems. While the goal of the FDM systems is to enable a proactive identification, FDM can only react to things that are in the data. The current state of the art identifies new safety risks through a series of unfortunate events. Creating a predictive capability that can identify risk areas before major incidents or accidents occur can enable truly proactive safety management FDM-type analysis could have identified the incipient issues in the Columbia SS accident, but only if the appropriate condition indicators had been put in place[56].

- Smaller operators and vehicles

  Because helicopter operators tend to be smaller in terms of number of aircraft and size of operation, as well as operate with lighter/smaller vehicles, the cost of HFDM systems can be too high for the operators on the small end of the spectrum. Further, the type of recorders and data acquisition units used by small operators may be less advanced than those used by the most fixed-wing operators and larger helicopter operators, which results in a reduced set of parameters that can be recorded. Since there is no defined standard for the type of recorders used on the majority of helicopters involved in HFDM, operators have to modify and re-define safety events to fit their hardware constraints. At the same time, many of the accidents are experienced particularly by the new/small operators[1].

1.4.2 Gaps

The challenges outlined in the previous section present a practical constraint on the successful operation of any FDM system, and HFDM in particular. "Safety events" form the basis for most algorithmic HFDM hazard detection, and the present review indicated that

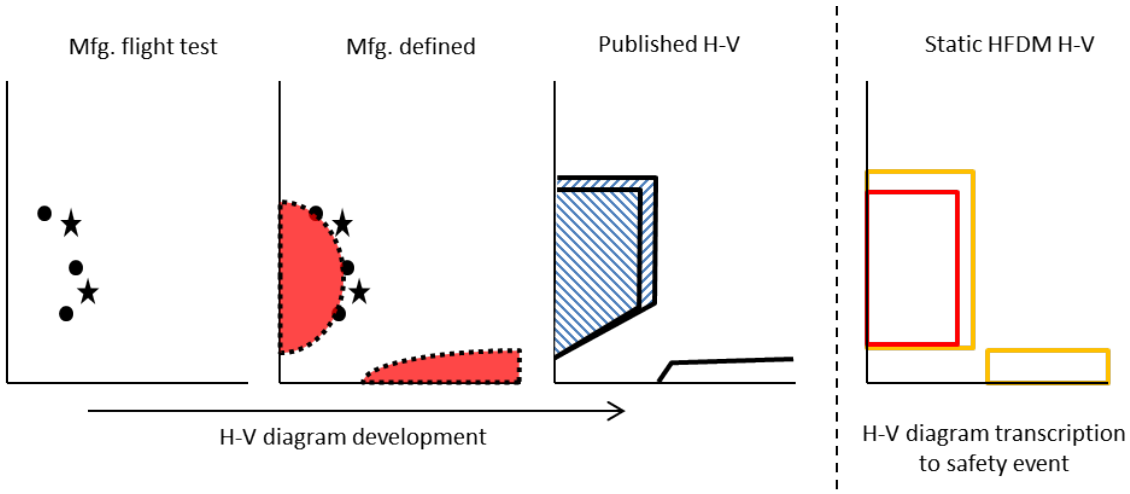


Figure 1.11: Notional process of developing an H-V safety event

a critical issue in HFDM systems today is the manner in which savety events are conceptualized, defined and implemented in practice. To illustrate the problem with the current process, consider the notional graph in Fig. 1.11 describing the progression of an event from manufacturer flight test through to implementation in an HFDM system. On the far left is a plot where a few points are shown, indicating the flight test points that had been investigated by the manufacturer for the vehicle in question. The manufacturer may run several iterations of this test, but the main point of this depiction is that after the tests had been conducted, the manufacturer obtains an estimate of where the safety envelope is located with a limited number of experimental runs. In this case, the notional graph depicts the H-V diagram, a critical limit for rotorcraft safety. This envelope is then published in a simplified form in the flight manual and other sources.

The safety events that are implemented within an HFDM system are further simplifications of this envelope. On the far right in fig. 1.11, the rectangular regions correspond to safety events defined using simple ”if-then” rules. The low speed ”avoid” region is defined by fixed thresholds on the height and speed parameters. Usually, multiple levels of severity are defined, so that the region in the far right image marked in red is more severe than the orange boundary. Overall, simplifications occur as the information passes from the flight test, through the manufacturer’s review and publication process, and finally through the op-

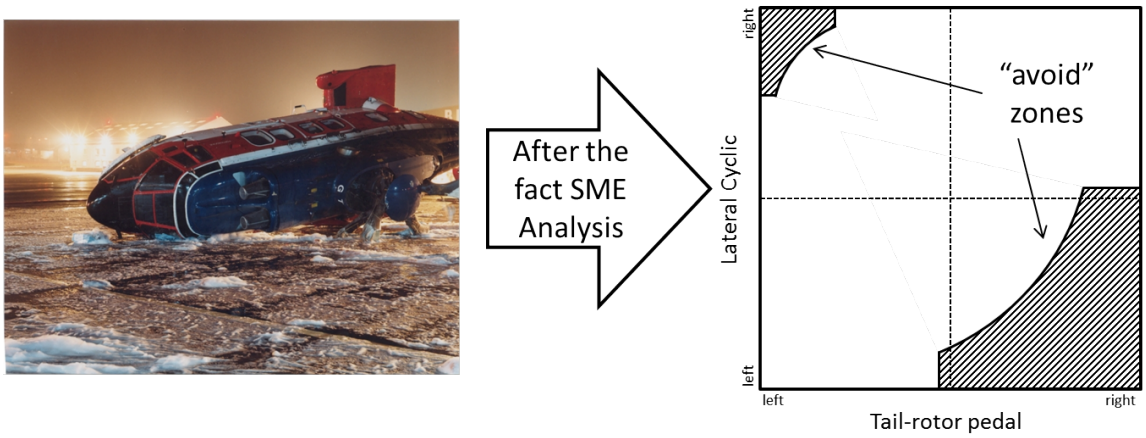


Figure 1.12: Safety event implemented *following* an observed accident[57]

erator's own SOPs and subjective HFDM safety event definition. The result is that HFDM events differ from the true boundary in the following ways:

1. HFDM events are defined in fewer dimensions, usually defined on a single parameter.

2. HFDM events are static whereas the original flight envelope varies with vehicle configuration and external influences.

3. HFDM events are not well established for improbable conditions

The final gap is a result of insufficient knowledge and lack of objective data surrounding rarely observed conditions. In HFDM, there is a tendency to neglect improbable scenarios because they had not been observed in the past, or by the specific operator. In such cases, an HFDM system offers no protection with the current system of event-driven detection. Yet such examples exist in practice, such as the often cited G-TIGT[57] accident shown in Figure 1.12. The helicopter belonged to an operator with an established HFDM system, and the only reason no detection was possible is the fact that an event had not been defined for this condition prior to the accident. Following this accident, a careful review by the manufacturer created new "avoid" zones for this particular helicopter model.

To remedy this deficiency, an improvement to the current state of the art is needed. Specifically, the new approach must address the differences between the true envelope and

the implemented limits. Further, a predictive capability that can identify limit boundaries based on the physics involved will help minimize false alerts and make the limit setting more objective. Such a capability could reduce the effort in finding and defining new safety events, as well as enable simpler transition of the established event set to different helicopters. The development of such a capability is the focus of this research, with a particular focus on techniques that operate within the constraints of a typical HFDM system, for the purpose of easing transition to an implemented solution.

Research objective: Develop a methodology that will improve HFDM detection capability through predictive and objective identification of safety limits while accounting for changes in operating conditions, characteristics of the monitored vehicle, and under constraints of typical HFDM data.

CHAPTER 2
BACKGROUND

This chapter is devoted to a review of the literature on flight data monitoring, where researchers have sought to reduce the limitations of the expert-driven approach through various means. Primarily, the surveyed research can be divided into data-based and model-based methods based on their use of the available data and the representation of the physical characteristics of the monitored system within the method. The following sections describe an overview of existing methods with the intent of identifying pertinent characteristics that might point to a clear avenue for the development of enhancements for HFDM.

2.1 Statistical learning approaches in flight data monitoring

With the increased availability of large data sets from various sources, the tasks of analyzing and extracting useful knowledge from these data has become a vary active research area. This process is known as Knowledge Discovery in Databases[58] and can be defined as "...applying data analysis and discovery algorithms that, under acceptable computational efficiency limitations, produce a particular enumeration of patterns (or models) over the data." In the literature the terms *data mining*, *anomaly detection*[59], *machine learning*[60], *pattern recognition*[60] and others seem to be used interchangeably. Because of the vastness and active nature of this field, it is quite challenging to obtain a concise statement of the boundaries that define it, as well as arrive at a conclusive taxonomy of its methods. Moreover, researchers tend to give new names to slight modifications of existing techniques after adapting them to their particular application, complicating the assessment of candidate methods for application with flight data[61]. For a fairly complete introduction, the reader is referred to some of the more comprehensive sources on the subject [62, 63, 64].

One of the most well-agreed upon distinction within the broad knowledge discovery field is that between *supervised* and *unsupervised* techniques. The distinction arises from the fact that certain datasets contain input-output pairs of values, while others only contain output data. In the context of HFDM, the inputs can be considered to be the flight conditions observed in the data, while the output would be the presence of an HFDM event. If such a dataset were to exist, and span the entire operational space, then supervised techniques could be used. In theory, once an HFDM analysis has been performed on a set of flight data records, a supervised technique could be used to "learn" the relationship between the flight data and the detected events, but there does not seem to be significant benefit to such an approach. A far more powerful application would be to use the raw data in order to identify natural structures within the data which could indicate the presence of anomalies. Therefore, most of the applications that use KDD techniques with flight data favor the *unsupervised* approach, which is reflected in the following review of representative anomaly detection studies.

Amidan[65] described the Aviation Performance Measurement System (APMS) project and an approach for anomaly detection based on Singular Value Decomposition (SVD). Two approaches were used, based on the features constructed: Routine Event (RE) and Time Interval (TI) methods. In the first approach, predefined conditions were used to create snapshots of flight data (the REs). The principal components of the normalized REs were used to identify outliers that would signal an anomaly. The second approach created features by fitting regression models to intervals within separate flight phases. The coefficients of those models were summarized and used as features on which principal component analysis is performed, as in the RE approach.

Mugtussidis[66] implemented a classification to differentiate between typical and atypical flight data, and used the Flight Data Clustering (FDC) algorithm for fast sequential data clustering. Instead of the direct Bayesian classification approach, which seeks to identify anomalous flights, he used a Bayesian classifier that identified a flight from which a

particular data point was obtained. A decision cost function allowed for the automation of the classification procedure and reduced human interaction. The FDC algorithm used a sequential clustering scheme to form compact spherical clusters. The method identifies samples that are close to each other and represent them with a center and a weight.

Iverson[67] developed the Inductive Monitoring System (IMS) for real time monitoring of complex systems. IMS used nominal data sets to extract general classes that represented typical system behavior, and later real-world data were compared with these classes to point out abnormal behavior.

Budalakoti et al.[68] implemented CLARA in the development of SequenceMiner - a tool used to detect anomalies in discrete symbol sequences based on the Longest Common Subsequence (LCS) measure. Their algorithm used both clustering methods and a Bayesian model. They applied the algorithm to 2200 flights consisting of the landing phase-sequence information. Some of the important discoveries made by SequenceMiner included inappropriate activation of the engine igniter switch, and mode confusion – a situation in which the pilot is not fully aware of the configuration of the autopilot.

Das et al.[69, 70, 71] developed a technique called Multiple Kernel Anomaly Detection (MKAD) for detecting abnormal instances within a large set of flight data records. MKAD allows the processing of both discrete and continuous data. Kernel functions are user-defined functions whose output represents the similarity between two objects. For each type of data, a single kernel function is defined, and all the kernel functions are combined to a single multiple kernel function, which makes it possible to deal with heterogeneous data simultaneously. This single function is optimized using One Class Support Vector Machine (OCSVM). The OCSVM uses a training set of nominal examples (in this case, flights) and evaluates test examples to determine if they are anomalous or not. After completing this analysis the algorithm reports the anomalous examples and determines whether there is a contribution from either or both continuous and discrete elements. A detailed discussion of the above work was reported by Das et al.[69]. They review several of the

previous works and conclude with comments regarding the use of data-driven algorithms in anomaly detection for aviation. One of the main conclusions reported by the authors is the importance of appropriate data pre-processing, transformation and normalization. Also, the techniques they reviewed make strong use of the fact that the analysis is being performed on a fleet of very similar systems, which is one of the primary assumptions in most of the reported techniques.

Other recently developed anomaly detection techniques for use with flight data are those by Li[72] and Smart[73]. Li et al.[72, 74] developed Cluster-based Anomaly Detection (Cluster-AD) to identify unknown issues from FDR data of routine commercial flights. This work considered only well-defined phases of flight, such as the takeoff and landing. For the landing phase, the final part of the flight was discretized at uniformly distributed distances from the landing field. The touchdown point was used as an anchoring point for the descent profile, with a radius of 6 nautical miles around the landing point used to define the initiation of the descent. This approach was justified by the fact that most airlines have descent profiles specified in terms of distance from the airport, and allowed a relevant metric for comparison to be created. Specifically, the vehicle height at the different distance points represented a feature vector which were then clustered using DBSCAN. Any outliers from the initial data set, and any abnormal data from new records, can be flagged as an anomaly and passed on for further analysis.

Smart[73] also used a data-mining approach to identify anomalies during the descent phase of an airliners flight. In his review of existing methods and the current FDM paradigm, the event-based detection was found to be deficient in that it only captures known issues for which definitions have been created. To extract more information from the available data, Smart used k-means, Mixture of Gaussians, and Support Vector Machine (SVM) techniques for outlier detection, using features created based on Subject Matter Expert (SME) input. The proposed method is a two-phase approach. In the first phase, the data are partitioned into training and test sets and features are constructed by taking snapshots

at pre-defined heights along the descent profile. For each height a classifier is trained, and for each altitude snapshot the distance to the classifier threshold is recorded. This distance is used to generate a Descent Abnormality Profile (DAP), using the distances to the classification boundary at each altitude for every flight. In the second phase, the DAPs from the training set are used to create another classifier, which is used to identify anomalies. Finally the anomalies are ranked in terms of distance from the classifier threshold (and in turn, in terms of severity).

2.1.1 Observations on statistical models

Statistical approaches include both traditional techniques, based on the review and analysis of accident reports, as well as more recently developed knowledge discovery algorithms. The former has been in use for a long time and forms the standard approach to monitoring safety over time, through various representations of accident and incident rates. The benefit of the traditional approach is that it does not require significant computational effort, flight data, or operator participation. The drawback is, of course, the completely retrospective nature of the analysis and reduced ability to generate timely indication of emerging safety concerns.

There is a number of studies that employ more advanced techniques for data analysis, such as data-mining and anomaly detection. Most of the research cites the problem of improperly defined apriori safety events as the primary cause for investigating purely data-driven alternatives.

A common assumption across most of the reported literature is that the data originates from a very uniform fleet of vehicles, with a similar operational profile. In some cases[73], the analysis was performed on a single aircraft type, during a particular phase of flight(landing), and at a specific landing location. It is unclear how well this type of analysis generalizes and allows comparison between flights conducted at different locations and with different aircraft, but the results reported when the uniform fleet assumption

holds seem to support the application of anomaly detection techniques to flight data analysis. However, the operational spectrum encountered in helicopter operations is much more varied compared to that of commercial airlines, and operators tend to have smaller and less uniform fleets of aircraft.

The anomaly-detection techniques generate alerts and prompt the system operator to investigate the anomaly. Thus a human expert is required within the analysis loop, but the set of automatically flagged conditions is a very small subset of the overall dataset. The benefit of the knowledge discovery techniques, especially the unsupervised anomaly detection algorithms frequently reported in aviation applications, is to generate this small subset and focus attention on potentially dangerous conditions, thereby improving the ability of the operator to detect and mitigate safety concerns.

A common aspect across most of the statistical approaches is that the techniques used operate on some derivative based on raw flight data, called *features*. Frequently, the temporal nature of the data can be used to relate parts of the data in a logical manner[68, 75]. The literature seems to agree in the observation that including domain-specific knowledge in the data-mining task improves the chances of identifying truly anomalous conditions.

2.2 Physics-based safety analysis

This section discusses physics-based approaches to helicopter safety analysis. More precisely, the review seeks to provide an answer to the question of how to reduce the reliance on expert-defined safety events. Physics based approaches are a common alternative to the purely data-driven techniques used in the field of system monitoring and fault diagnosis[76, 77]. Some of the advantages of including the physics of the problem into the monitoring task include:

- Ability to relate measurements to unobserved quantities which could be better suited as a detection signal

- Allow comparison between simulated and measured output
- Predictive ability to assess system states outside the collected data, but within the domain represented by the modeled physics

The two primary paths to creating a physics model of a helicopter are through system identification and through first-principles modeling, depending on the available information and the desired use of the model. Below is a brief review of relevant work in both directions, starting with a discussion of the purpose for which the models used in this method are being developed.

2.2.1 Model requirements

The previous chapter introduced HFDM and discussed the limitations in current practice. It was shown that the main cause for reduced HFDM capability, and in turn reduced safety, is the reliance on apriori safety event definitions. This chapter introduces related research into techniques that can mitigate the problem posed by inaccurate or non-existent safety events. The reviewed literature can be grouped into statistical learning approaches and physics-based modeling techniques. Physics-based modeling techniques are further divided into system identification and first principles techniques. It is shown that due to the constraints inherent in HFDM data streams, the first principles modeling approach is a natural first step to resolving the knowledge gap that hinders current HFDM-based safety efforts.

Safety conditions of interest

During the course of the preceding literature review, it became apparent that any of the potential approaches to circumventing existing HFDM events would require some degree of physics-based modeling. For these models to accurately represent the vehicle in a safety-related situation, the effects which contribute to safety degradation must be captured. This section first discusses relevant limits on a typical helicopter, and presents potential can-

didate techniques for consistent evaluation of rotorcraft operations near these undesirable conditions.

Known limits and hazards in helicopter operations

A good source of limit boundaries is the flight manual for a particular helicopter[30] and the FAA Helicopter Pilot Handbook[50] for general helicopter hazards. Within these sources flight envelopes are usually specified in using the following quantities:

- Attitude and airspeed limits
- Speed and load-factor boundaries
- CG limits
- Height-velocity curves
- Component limits (main rotor RPM, torque)

The limits documented in operating manuals are usually a simplified subset of the actual boundaries experienced in flight, and frequently contain some (potentially subjective) margin between the values determined by the manufacturer, and those contained in the manual and communicated to the operator. Whalley et al.[78] conducted a survey of the limits that pilots must avoid during operation and found upwards of 40 different limits, including limits on the ground, in the air and during takeoff. Most of these limits are related to component loads, Loss of Control - breaches of manufacturer envelope, and ”handbook limits” like the H-V curve which only manifest themselves during engine failure or other malfunctions. Most of the limits mentioned in this study vary with flight condition and vehicle configuration, which complicates the pilot’s task. Consequently, pilots tend to adopt either a conservative attitude and under-utilize the capability of the helicopter or expose the flight to potential dangers with frequent excursions beyond the ”handbook” limits [79].

There are also limits which are difficult to define prior to an accident or well-documented condition that reveals the mechanism in which the limit affects safety. A frequently discussed example in HFDM materials is the rollover during taxi - where an event was defined to capture ”near rollover” conditions only after an accident resulting in damage happened[80, 57]. After a safety event was defined and implemented, the analysis revealed that the occurrence was not an isolated incident and that many more instances of similar conditions have happened both before and after the rollover. This points to the fact that even for known types of failures, an unexpected combination of conditions can lead to a previously unobserved accident.

It is apparent that helicopter limits arise from a wide range of effects and capturing them all would require a full dynamic model with a complicated structure. As a means of managing the development burden, it is desirable to obtain a reduced set of limits which are known, but still result in a large number of accidents. In the survey of U.S. helicopter accidents, it was found that over 70% of all LOC-type of accidents were due to performance mismanagement and dynamic rollover[81, 8]. Of the performance mismanagement accidents, the primary cause was found to be improper autorotation technique in both forced and practice situations. Rao et al.[82] performed an examination on a more recent set of accident reports. Their findings reflect similar trends as to the criticality of the dynamic rollover and autorotation to safety, with most accidents deemed ”unrecoverable”.

In this work, we consider several conditions which have contributed to many helicopter accidents, particularly *loss of control* accidents. Studies of helicopter accidents[81, 83] have revealed the main categories of accidents to be related to the pilot's operation of the vehicle. The goal is to address the more significant areas of concern first, and the measure of the significance of an accident type has historically been the number of such occurrences over a period of time. As discussed in [84], the accident rate as a metric is not ideal but provides a direction for the focusing of development efforts on more proactive metrics. The industry is in general agreement about the types of safety concerns that helicopters are

most commonly exposed to, and these known hazards provide the starting point for this investigation.

Of particular interest is the management of performance during normal operation and during emergency procedures such as autorotation, dynamic rollover, loss of trail rotor effectiveness and entry into the vortex ring state, among others. These hazards are frequently discussed in rotorcraft literature, such as the Helicopter Flying Handbook [50]. Below are brief summaries of the conditions considered in this work, and for more details the reader is referred to the handbook and similar literature. **Autorotation** is a flight regime in which the rotor is powered by the airflow through the rotor disk. Portions of the rotor produce driving torque which maintains the rpm, and keeping the helicopter at the proper attitude requires deliberate input by the pilot. Since pilots have only one opportunity to execute a safe landing, extensive practice is needed to master this maneuver. This fact exposes pilots to the hazards associated with improperly executed autorotation both during practice and actual emergencies.

The **Vortex Ring State (VRS)** is a well-known aerodynamic condition that is encountered at moderate forward velocities and descent rates that are roughly 0.5 to 1.5 times the value of the hover induced velocity through the rotor disk. In VRS, the flow enters a recirculating pattern, which has the effect of reducing thrust and causing an increase in descent rate. This condition is especially hazardous close to the ground while operating at high altitudes above sea level. Another concern with VRS is that the typical response of the helicopter is reversed, such that collective application may increase the descent rate. The primary recovery method is to fly away from the recirculating flow.

Dynamic rollover is another hazard of interest, where the helicopter rolls over while on the ground due to the combined action of the main and tail rotors with a pivot point established by the landing gear. While dynamic rollover is a well-known hazard, its name hides the fact that there are several possible ways to encounter a rollover, both dynamic and static. In dynamic, the vehicle is translating or lifting off, while the landing gear

provides a pivot point when it is prevented from moving. A static rollover is primarily experienced on helicopters with a high-mounted tail rotor, where the increased moment about the ground contact point can cause a tipover. In either case the recovery requires pilot inputs that may be counter-intuitive, as cyclic control alone is usually not sufficient to prevent a rollover. Further, during normal flight an increase in collective input also increases the control moments possible through cyclic input. While in contact with the ground, the increase in the moment about the pivot point far outweighs any gains in control forces and moments at the rotor, so this typical pilot response can be counterproductive in rollover prevention.

Thus the present investigation will use these hazardous conditions to focus the development of the methodology, with the goal of identifying models which can provide additional information and make relevant predictions for the conditions described in this section.

2.2.2 System Identification approaches

Most of the approaches described in the section on knowledge discovery techniques make extensive use of the flight data in order to detect anomalous conditions. In most cases, the researchers agree that some transformation of the raw data is required to enhance the analysis (feature generation/extraction). In some cases these features are constructed in a way that uncovers salient properties of the system, in effect incorporating physical characteristics of the aircraft into the analysis. For example, Gorinevsky[85] used flight data from a fleet of commercial airliners to train a performance model with a known structure for each vehicle represented in the fleet. The study made use of a half million flight records during the initial investigation, and was applied to the cruise segment of the flights. The researchers cite three forms of monitoring in their approach:

1. Residuals between fleet-model generated output and incoming data for individual flight.

2. Residuals between fleet-model and long-term measurements on a particular aircraft

tail number.

3. Mismatch between fleet-model parameters and model fit using data from single aircraft.

The type of work conducted by Gorinevsky is traditionally the realm of System Identification[16], though only the performance aspects were considered.

In their broad review of System Identification (SysID) techniques for aircraft, Klein and Morelli[16] define SysID as "... the determination, on the basis of observation of input and output, of a system within a specified class of systems to which the system under test is equivalent." They explain several key aspects of SysID implicit in this definition:

- Mathematical models generated through SyID are not unique.
- SysID is based on observations of both inputs and outputs to the system.
- In practice, measurements of inputs and outputs are corrupted by noise, requiring the application of statistical methods
- The equivalency between the observed system and its mathematical representation must be defined, for which there are multiple approaches.

Though most of the techniques for SysID require the implementation of statistical estimation theory, SysID is discussed here as separate from the purely data-driven approaches of section 2.1 due to the explicit consideration usually given to the physical characteristics of the system, such as kinematics used for consistency checking. The techniques employed in SysID can be executed in both the time-domain and the frequency-domain, depending on the problem under consideration. Another important reference on system identification techniques, with a particular focus on rotorcraft, can be found in the work by Tischler et al.[17]. In an early paper[86], Tischler describes the general frequency-domain identification and time-domain validation methodology. It is shown that valid models can be extracted with excellent accuracy from flight experiments with rotorcraft. Similar work is

reported in Hamel et al.[87]. For a broad review of the development of SysID techniques over a period spanning over three decades, see Gevers[88]. The application of SysID methods with HFDM data is attractive for its potential to create a model from recorded data. The identified model is usually considered a local approximation, and has some predictive capability that diminishes away from the condition used in the identification. Within the region of validity, a model developed using SysID could be used to gain additional understanding of the behavior observed in new incoming data, potentially allowing unobserved quantities to be estimated and used in the safety analysis.

The system identification approach relies on the availability of adequate input-output data for the system that is being modeled. Some desirable features of the data are high sampling rate, sufficient content at the frequencies of interest, and samples from a broad range of operating conditions. In structured system identification trials for aircraft, a necessary step is the definition of the maneuvers to be performed. Some of the most common maneuver types used in aircraft system identification tests are doublets, chirps and frequency sweep inputs. Figure 2.1 shows a representative frequency sweep input. During this type of maneuver, a control axis is chosen and the input is manipulated as shown in the upper portion of the figure. The goal of this type of maneuver is to excite a wide range of dynamic response frequencies, which increases the information content of the data and aids the subsequent identification.

However, the flight data commonly recorded in HFDM systems presents some challenges which complicate the application. Specifically, the recording rate and frequency content is very different between instrumented flight tests for the purpose of generating usable system identification data, and the HFDM data generated during regular operations. The low sampling rate creates the possibility for problematic aliasing, and the low frequency content of the flight data limits its application to largely steady-state conditions.

An alleviating condition is that for most applications, a model form can be postulated and then the problem reduces to the estimation of the parameters within the model. *Pa-*

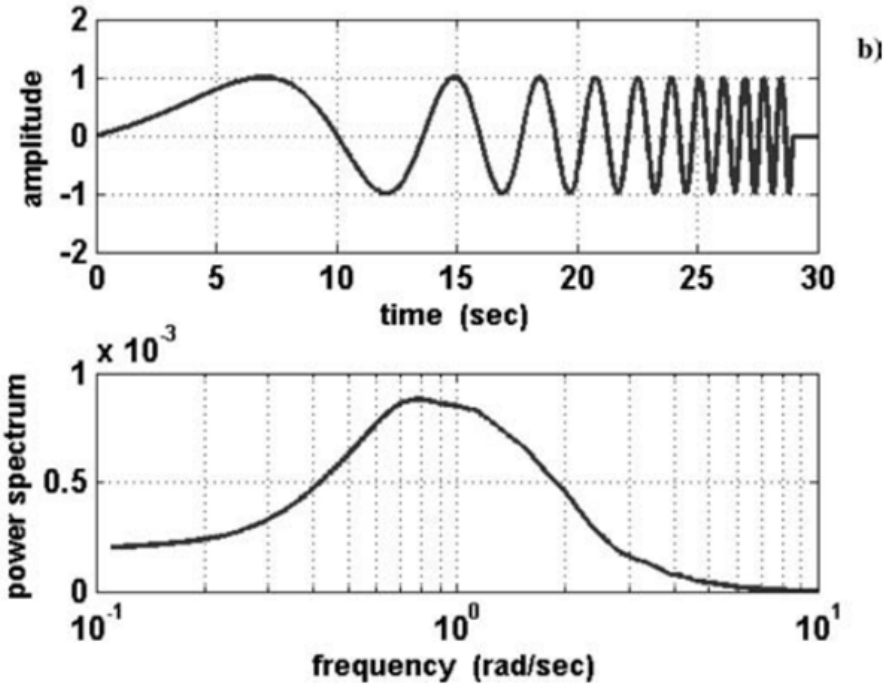


Figure 2.1: Power spectrum of frequency sweep input[16]

Table 2.1: Flight data requirements for SysID vs. HFDM data properties

Characteristic	**SysID**	**HFDM**
Recording Frequency	>50Hz	1-4Hz
Maneuver type	Sweeps, chirps, doublets	Mostly steady
Data type	Input-output, highly instrumented	Usually output only

rameter estimation is a subset of the overall SysId framework, and reduces the burden of obtaining high-quality data. Some of the more common approaches are the equation-error and the output-error methods. In theory, both should be applicable with FDM-type data, provided the data contain input-output pairs and contain sufficient information, though the output-error seems to be preferred in practical applications[89]. Sembiring showed an application where Bayesian inference was used with the data to estimate model parameters of a simplified aircraft deceleration model[90]. The identification focused on three model parameters, specifically the C_L,C_D and T (lift and drag coefficients, thrust), during the approach and landing phase of an airliner. The input data contained control deflections for

thrust and flaps, which were included in the model and accounted for changes in the lift and drag coefficients. The analysis was performed by first separating the flight into phases which facilitated parameter estimation of individual models. The optimized model shows much improved correlation with the measured data.

In addition to the problems arising from the low sampling frequency and dynamic content in typical HFDM data, the flight data records usually contain very few records of the conditions at the boundary between safe operation and unsafe occurrences.

2.2.3 First principles models

The alternative to system identification methods is to construct a model from first principles, where the vehicle is modeled with explicit formulations for every relevant component. This section describes some of the possible approaches to developing a helicopter model without flight data requirements. First, steady state performance models are discussed. Dynamic models are reviewed next, with consideration of the fidelity as a function of the application.

An operating assumption when creating models from first principles is that sufficient information about the vehicle is available to create the model. In the context of HFDM, the models represent helicopters currently in service, with extensive documentation available.

For simple evaluations an energy-based method may be used, such as the momentum theory model described in many well-known helicopter references[91, 92, 93]. The power required in hover is given as a function of the thrust T and the inflow $v_i = \sqrt{T/2\rho A}$. In the ideal case this is simply $P = Tv_i$, but in reality there are non-ideal losses due to blade drag:

$$P = T\sqrt{\frac{T}{2\rho A}}\frac{1}{\mathrm{FM}} \tag{2.1}$$

where FM is the *figure of merit*, an empirical correction to account for profile losses not considered in the derivation of momentum theory results. A is the rotor disk area calculated using the rotor radius πR^2.

The momentum theory provides good first estimates of the energy balance at the rotor and is useful for performance calculations, and even some basic consideration of steady-state maneuvers[91]. A common assumption at this stage is uniform inflow through the rotor disc. The derivation of uniform inflow neglects any rotation imparted on the flow by the rotor, and assumes a well-defined stream tube where the wake is localized in order to apply the conservation of momentum principle. This approach can be applied in forward flight if the inflow calculation is extended to include the flow through the disk due to the forward velocity of the helicopter and the tilt of the rotor. Accounting for fuselage drag, additional profile losses in forward flight, the contribution of the tail rotor as well as any climb velocity results in the following equation for the power coefficient in forward flight:

$$C_P = \frac{kC_T^2}{2\sqrt{\lambda^2 + \mu^2}} + \frac{\sigma C_{d_0}}{8}(1 + K\mu^2) + \frac{1}{2}\left(\frac{f}{A}\right)\mu^3 + C_{P_{TR}} + \lambda_c C_T \qquad (2.2)$$

This equation is the basis for many performance analyses, and can be utilized in simplified steady maneuver analyses. In initial analyses, the above equation can be rearranged to yield the descent velocity required in autorotation. Thus it lends itself to investigations of this flight condition, though only as a first estimate. For non-steady maneuvering, a dynamic model is necessary. The literature on dynamic models used in helicopter evaluations contains a wide range of models, from point mass models through to full non-linear models with rigid 6-DOF fuselage motions (with additional degrees of freedom for the rotor, inflow, and any other components). This type of model corresponds roughly to the Level 1 model as described in [94] and is further detailed in the description of the proposed approach. Models that capture higher-frequency motions also exist, but are not frequently used when the global helicopter motions are of interest.

The construction of a helicopter H-V diagram has been investigated using dynamic models due to the fact that the flight testing required to generate the necessary data re-

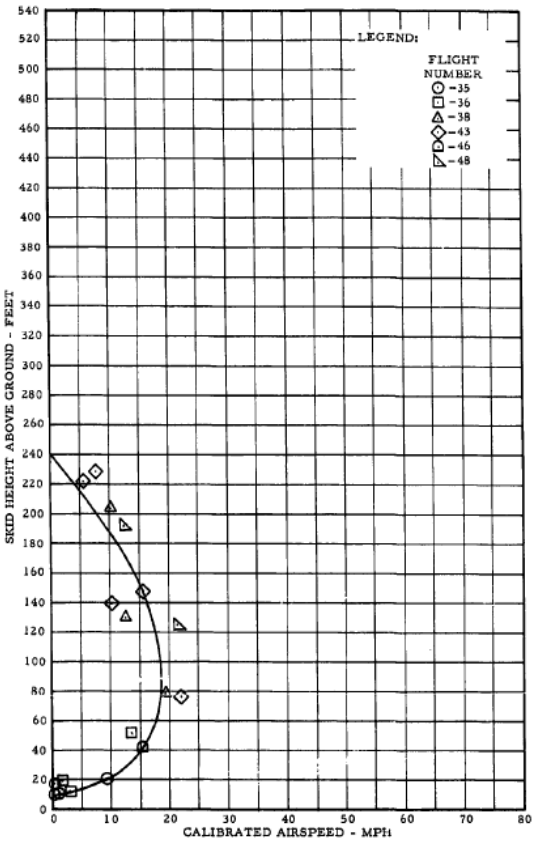


Figure 2.2: Typical single-engine helicopter H-V diagram[95]

quires the pilots to approach conditions which are inherently dangerous and may result in significant damage. The H-V diagram defines the boundaries of a critical region where autorotation cannot be performed safely, and is approached by performing autorotations from outside the region. Each consecutive flight test is started closer to the boundary until the pilots determine the boundary has been reached[95, 96, 97, 98]. Simulating the same maneuvers as the pilots would perform in flight reduces the burden on flight test and could provide information about the H-V of a particular design before it is even flown. This usually requires a fairly detailed model, but since the entry into autorotation is primarily a longitudinal maneuver, the number of degrees of freedom can be reduced to only include the relevant motions, thereby simplifying the analysis. This approach was taken by Okuno et al. [99, 100] in their analysis of engine-out landings using a 4-DOF model (translation and rotation, and rotor speed). Bibik analyzed the autorotational performance of a PZL

Mi-2 using an 8-DOF model with rigid-fuselage, rotor and engine dynamics. Deresz[101] used the proprietary flight dynamic simulator FlyRT to evaluate the H-V diagram of an attack helicopter under a variety of conditions. These studies usually rely on optimal control methods in conjunction with the dynamical model to investigate the suitability of various flight conditions from a safety perspective. Their results are in general agreement with prior published results, in terms of identifying the shape and size of a H-V diagram or an autorotative boundary given a description of the helicopter.

In addition to the mathematical model of the vehicle under investigation, these studies rely on the definition of an appropriate cost function which is minimized in the optimal control framework. Bottasso et al.[102] showed the potential for solutions to differ when a reduced number of degrees of freedom are considered. The difference found in the study was relatively small and may in fact point to the benefit of using limited models if the computational and analytic effort is significantly reduced. A full model is still preferable if the scope of the study allows the additional burden of model development, or if a model is readily available.

2.2.4 Observations on vehicle models

The review of physics based methods included system identification and first principle methods. System identification methods were found to require properties which the HFDM data presently available to this research do not posses. Specifically, the absence of control inputs in the data make it problematic to apply a system identification technique.

In the case that input data become available, the system identification sub-task of parameter estimation seems to have the highest probability for success when applied in an HFDM setting. This will be revisited if such data become available.

For the modeling aspect, the choice of model depends on the types of characteristics and the physics of interest, and the types of conditions that contribute significantly to accident rates. It was found that autorotation and dynamic rollover are responsible for a high propor-

tion of all LOC-type accidents. To study the basic aspects of these conditions, simplified models can be used to capture the relevant physics and yield an improved understanding of the actual safety limits. In steady state, performance point models may be a valuable addition to the current HFDM state of practice. If dynamic conditions are of interest, again simplified models with a reduced number of degrees of freedom and fewer effects may still offer an improvement over the current trial-and-test approaches in HFDM.

To capture most of the effects important to a safety analysis that is not limited to a particular maneuver of flight condition type, a full dynamic model is required, with an appropriate level of detail in terms of physical effects. Since the majority of the conditions of interest pertain to LOC, such a model would correspond to the type described as Level 1/2 by [94], depending on the formulation of the rotor flapping dynamics and inflow.

In this work, the application of several models is investigated, progressing to a nonlinear dynamic model that can be expected to provide reasonable results at the conditions investigated in simulation. In particular, FLIGHTLAB® will be used as a general "black-box" model in addition to any models developed specifically for a narrow operating condition.

2.3 Detecting the proximity to a limit parameter

If the limits associated with reduced safety are to be related to an observed flight condition, some measure of the distance that is relevant to the immediate safety of the vehicle is needed. In the simplest form, this can be the proximity of the current flight condition to the published flight envelope, which would indicate a potential for LOC. Belcastro et al.[103] identified these type of LOC metrics as highly dynamic, dependent on the vehicle physics, and arising out of effects that are not encountered in most of the flight envelope. Based on this information, the limit metrics which are of particular interest in this research are those which could be used alongside a physics-based model to detect when the known flight envelope is approached, or otherwise have a way to detect/estimate the proximity of a physical limit to the current state of the helicopter.

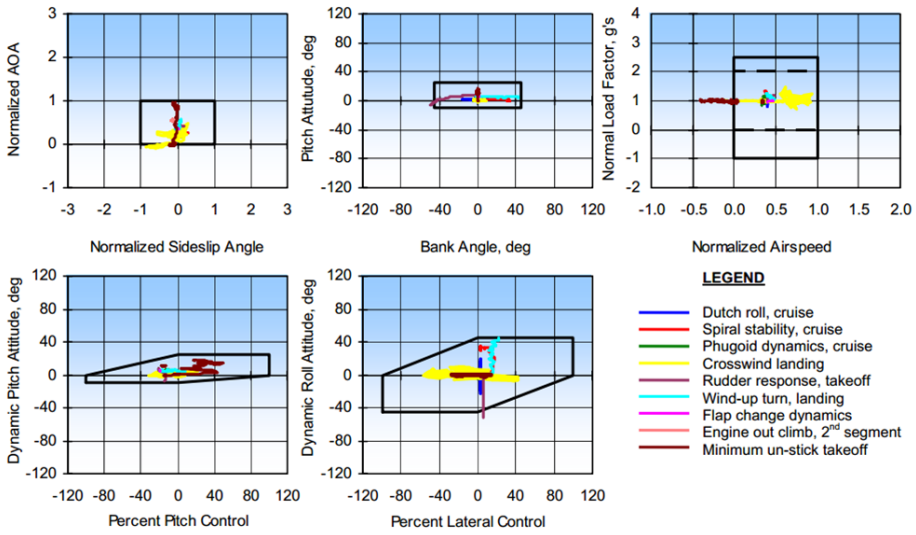


Figure 2.3: Limit envelopes for commercial transport[104]

Wilborn[104] introduced five limit metrics for a transport aircraft related to flight dynamics, aerodynamics, structural integrity, and flight control use. Two of the five envelopes contain a dynamic formulation, while the rest are normalized versions of flight manual limits or FAA regulations. The normalization was important in order to allow comparisons between different aircraft (that are trimmed at different attitudes). Plotting flight data against these envelopes allows one to detect when the flight condition lies outside what is considered safe. This approach is the currently used approach in HFDM, though the implementation is static and does not allow safety events to be applied to data from different helicopters. Angular position was also used in the investigation of rollover conditions by Fox[105].

Control-based metrics have been developed in the envelope protection and carefree maneuvering community. Envelope protection methods either implement the protection into the control signal, or alert the pilot of the impending breach[106, 107, 108]. The latter is known as limit cueing. A signal is generated that alerts the pilot of an impending limit parameter, giving the pilot time to respond to the situation. Another approach is to implement the alert through tactile cues, where the control stick is vibrated or the force feedback is modified to give the pilot a sense of the proximity to the limit. In these cases,

the limit is fairly well known, and is mapped to control inputs through the equations of motion for the system and the evolution of the limiting parameter. The mapping is quite complex and is frequently accomplished using a neural network model. Once available, the mapping can be used to relate the future breach of the limit parameter to the instantaneous control inputs, whether to modify the control inputs or to provide an alert to the pilot.

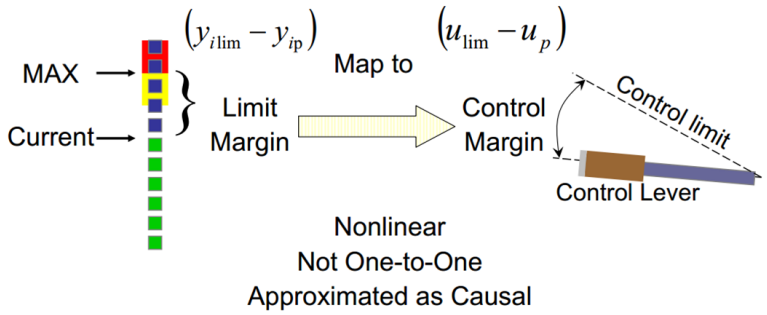


Figure 2.4: Limit detection and cueing[107]

Energy based metrics have been investigated in the past for fixed-wing and rotorcraft[91]. In fixed wing aircraft, the energy state creates a relationship between the current altitude and the ability to ”zoom” to a higher altitude by trading kinetic energy for potential energy in the case of an engine failure. For rotorcraft, a distinctive difference is that a portion of the total energy is stored in the rotor and the relationship between kinetic and potential energy is not as straightforward. In rotorcraft applications, it is important to know or estimate the amount of energy stored in the rotor to assess safety using this type of metric.

In the fault diagnostics and prognostics community, the focus is on detecting when an undesirable condition has been achieved by the system under observation. For example, in a rotating machine part, excessive vibration or noise may signal a malfunction, and a model of the fault can be used to estimate the progression of the fault and the ultimate failure of the component[44, 109]. One of the common metrics used in this field is the Remaining Useful Life (RUL), or the likely length of time before failure occurs.

For the conditions that will be investigated in the initial development effort, the energy

metric for autorotation and an angle-based limit parameter for rollover are appropriate as instantaneous limits. The known value of these limit parameters will be used to assess the merit of a general limit parameter which can be defined for most LOC-type limit variables. In the preceding review, a temporal component is implicitly or explicitly incorporated in most of the safety metrics. The energy-based metrics can be formulated as the rate of change of the energy state to detect the time required to reach a prescribed level. For the H-V boundary, a time-delay is specified to account for pilot action time[50]. In rollover detection schemes for ground vehicles, time-to-rollover is a frequently used metric[110, 111]. Most of the envelope protection field uses a receding time horizon estimation to predict future breaches of a limit parameter.

Time-based limit formulation naturally considers the dynamics properties of the system. In the case of helicopter simulation for safety analysis, this means that the vehicle can be excited arbitrarily and the response can be monitored for failure conditions. Additional benefits to this formulation is that the result will retain the temporal component. One of the main benefits of using the time-based formulation is that the results could be transcribed into a limit boundary for a given time-to-condition, thus forming an objective means to define HFDM safety events.

CHAPTER 3
PROBLEM STATEMENT

3.1 Research objective

The review of Helicopter Flight Data Monitoring state of practice illustrated the early-warning ability of such data-driven system. The helicopter industry is placing increasing pressure on operators to improve safety, and HFDM is a key enabler for that improvement. The review also uncovered several aspects of HFDM which currently limit the performance of such systems. Of the identified issues, the retrospective nature of HFDM due to static, inaccurate or undefined events was determined to be largest detriment to HFDM detection capability.

Research objective: Develop a methodology that will improve HFDM detection capability through predictive and objective identification of safety limits while accounting for changes in operating conditions, characteristics of the monitored vehicle, and under constraints of typical HFDM data.

The development of such a methodology should take into account the characteristics of the problem domain and make use of all available information. The following can be assumed about the HFDM problem domain:

- **Flight data are available** - Onboard recorded flight data are available for use in the development/application of the methodology. To keep the application domain as broad as possible, the minimum recommended set of parameters is assumed: GPS positions and velocities, orientation and accelerations data, recorded at frequencies typical of stand-alone HFDM recorders.

- **The helicopter model is known** - The type and characteristics of the helicopter being monitored can be obtained either within the data record or through other means. Thus published information about the particular helicopter type can be utilized in the development/implementation of the method.

- **HFDM performance is measured in terms of false alerts and missed detections** - The performance of the baseline system is known in these terms and will be used to evaluate the developed methodology for the improvement increment.

3.2 Research questions

Research Question 1: How can the available information in the HFDM problem be leveraged to avoid the pitfalls associated with static, a priori, simplified thresholds?

The literature revealed two potential approaches to reducing the reliance on typical HFDM events. The first approach is comprised of statistical learning techniques that operate on the flight data. There is a need in HFDM and other related monitoring systems (FDM, HUMS, etc.) to have established exceedance thresholds for the system to perform its task. Without these events, the typical HFDM system can not detect known safety issues and alert the operator, which means that any assessment of the level of safety of the recorded flight would be performed manually. For this reason, there has been some recent research into reducing the need for apriori event definitions and instead use the properties of the data themselves to identify unusual flight conditions, which may indicate a lack of safety.

A fundamental assumption in many data mining approaches is the existence of a large number of flight records, collected from a uniform fleet with a well defined operational profile. This allows the method to establish a baseline and detect any deviations from it. In helicopter operations, this may not be the case. Another caveat of the statistical learning approaches is that they output results which are physically meaningless, unless reviewed

and made sense of by an expert. Normalizations, transformations, and other numerical manipulations have a great effect on the results. In fact, the need to construct **features** on which the data mining algorithms operate have restricted the use of data mining to a few flight conditions, such as take off and landing, for which well-defined profiles exist (at least for airliners, which have been the focus of these studies).

These problems with data-mining approaches are exacerbated in the context of helicopter operators that can have a much more varied operational profile, with the associated lack of a well-defined baseline. Further, the data mining approaches do very little about conditions which have not been observed in the data.

To create a capability that can predict the response of the system (the helicopter) away from points observed in flight, it is necessary to move to a more **physics based formulation**, which will take into account all available knowledge and allow improved use of the data. Helicopter models have been used for many applications, and if an appropriate model is used in this methodology, the conditions of interest can be examined in detail and the appropriate limits set for use in HFDM.

Research Question 2: What are possible modeling approaches that can be used to create a predictive capability to investigate the envelope and revisit conditions experienced in flight?

The physics of the helicopter when operating near dangerous conditions needs to be captured for this method to enhance the detection ability of HFDM systems. The model needs to provide an accurate representation for the helicopter response, especially if the conditions of interest are in an unusual or poorly modeled region. The literature review in the prior section revealed two main approaches that are usually used to create a physics model of a system. System Identification is the first paradigm. It requires input-output data of various fidelity, depending on the purpose of the identification and the degree to which the physics of the vehicle in question have been specified. The second approach

is to generate the majority of the model through a component-wise build up and explicit formulation of physical effects. Of all the modeling approaches, this one is the most labor-intensive, but has the highest potential for generalization and is the only choice for the present research due to the constraints on the available data. The SysId and statistical learning approaches are still potential answers, but in order to make them successful, a representation of additional knowledge of the vehicle is needed. Pursuing the application of a first-principle model is seen as a necessary first step if the parameter estimation and statistical learning techniques are to be revisited.

The review and summary of the first research questions points to the direction that will be taken in this work. Since physics-based modeling and simulation is a clear direction given the constraints and needs of the typical HFDM system, the next chapter deals directly with the development of a methodology for physically relevant definition and detection of safety events in flight data from helicopter flights. The ultimate test for the proposed methodology will be a direct comparison with current HFDM safety event detection. The overarching hypothesis can thus be stated as follows:

Overarching Hypothesis: The model-based method of evaluating safety events would provide complementary information for traditional HFDM detection and aid in the reduction of false alerts and missed detections through better approximation of the true limit boundary.

This translates to improved performance from an HFDM system. One particular aspect that should be enhanced is the agreement between the monitored condition and the associated detection threshold. Another is the fact that most HFDM events are defined after an accident or high-risk situation is observed - so an approach that could identify the monitoring limits either in a predictive fashion or at least with reduced need for rework is preferred. If these capabilities are enhanced, the resulting system would rely less on analysts and their effort to create and maintain detection thresholds, and would consequently

increase the potential of correctly identifying hazards in real operations. The following chapter will describe additional questions and formulate hypothesis regarding the methodological choices with the goal of generating improvements in the performance of a typical HFDM system.

CHAPTER 4
METHODOLOGY

4.1 Overview

The primary purpose of this work is to devise an approach for the identification of safety critical flight conditions in rotorcraft operations and for the consequent definition of monitoring thresholds or events for implementation in an HFDM system. The requirements are that the physical mechanisms responsible for safety-critical events can be modeled and simulated, that the model/simulation can be executed in a manner that reveals these safety hazards, and that the output provides some measure of proximity to the undesirable state so that occurrences with higher risk can be communicated as such to the operator. The review of existing approaches pointed toward the need for physics-based models as a potential means of addressing the gaps in current HFDM practice and developing the missing capabilities. Two main uses of models of the vehicle physics are envisioned. One is a direct addition to the monitoring apparatus in the form of derived parameters that capture additional aspects of a particular monitored condition. If these models can improve the approximation of the true boundary, then the false alarm rates and missed detections should be reduced. A less straightforward but potentially more powerful approach is to not limit the investigation by devising condition-specific model, but use a generally applicable model which will yield a measure of the risk at a broad spectrum of applicable conditions.

The proposed methodology is coined **DEMMoS** - **D**efinition of **E**vents for **M**onitoring through **M**odeling and **S**imulation. Figure 4.1 shows a conceptual overview of DEMMoS as it relates to existing HFDM systems. The leftmost portion of the figure is a simplified representation of how HFDM systems operate presently. To operate, an HFDM system gathers data from regular helicopter flights. These data sets are then scanned for events in

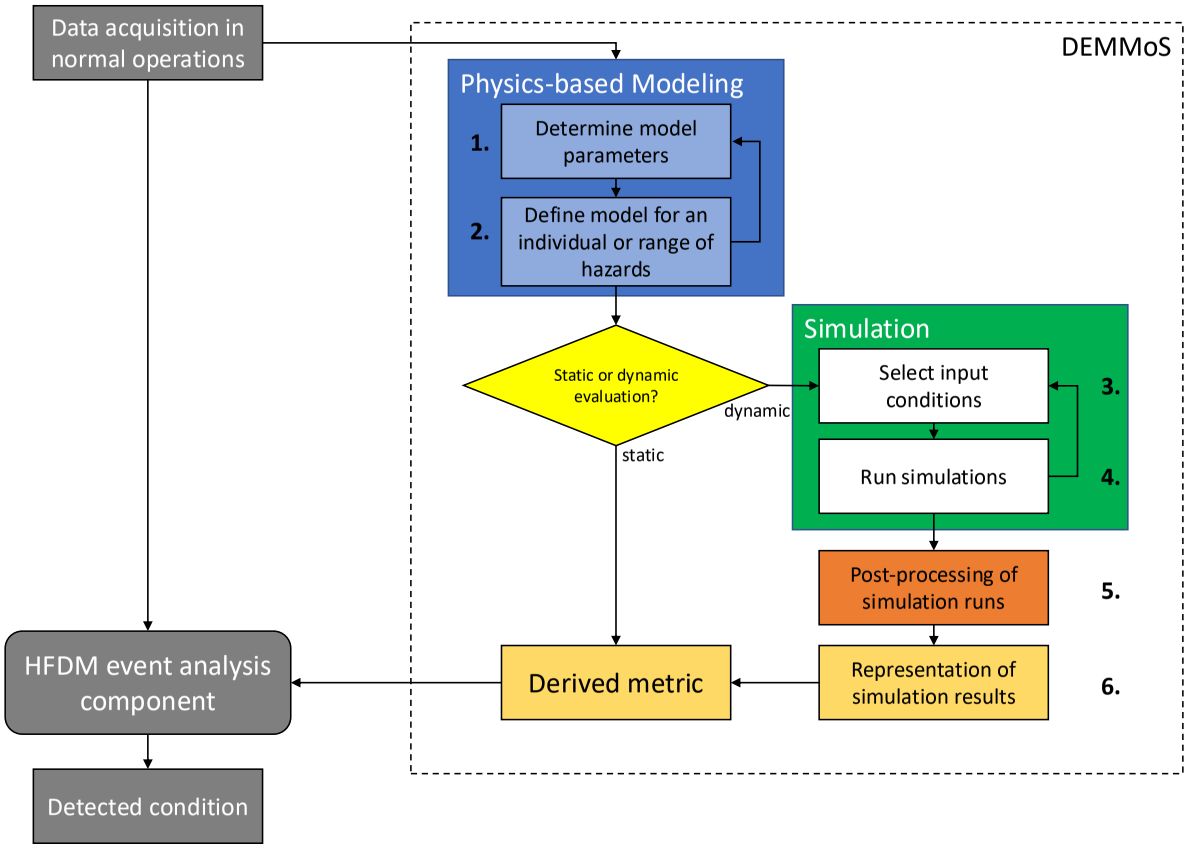


Figure 4.1: Overview of the method concept

the analysis portion of the system. When the data contain adverse flight conditions as defined in an expert-generated event database, a detection is indicated and the user can pursue further mitigating actions. The system then continues monitoring the operational improvement resulting from any such action. The model-based approach would complement the existing approach by providing both detection ability and the potential to proactively investigate underrepresented areas of the operational envelope. The DEMMoS methodology uses the physics of the vehicle in flight to derive "events" in order to help detect adverse flight conditions. The steps of the methodology that would allow such indicators to be defined are as follows:

1. Determine model parameters

2. Define model for condition(s) of interest

   The first two steps make up the modeling portion of the method where the physics of the vehicle are represented mathematically. This portion is repeated whenever the model is extended or implemented for new flight conditions if they warrant the formulation of additional physical phenomena. Depending on the extent and evaluation type of the model, two possible paths are possible. If the model is defined in such a way that its evaluation would directly result in a parameter which can be included in the HFDM analysis system, the parameter is included as an additional metric in the event database. On the other hand, if the model requires longer evaluation or cannot be executed in a single pass, such as is typically the case with a dynamic model, then the method requires separate simulation and other additional steps.

3. Select input conditions and simulation variables

   The selection of the input condition and simulation variables determines which conditions, and over what range, will the derived metric be applicable.

4. Run simulations

A simulation is run for every combination of input parameters, and the result is saved for later post-processing. Over time, repetition of the simulation at numerous flight conditions would yield a rich set of simulated data which would allow analyses to be performed in regions of the operational space that are otherwise poorly represented.

5. Post-processing of simulation results

   The simulation results are rarely in a form that can be immediately implemented, and some post-processing is required. Specifically, a relationship between the input domain and the safety observed in simulation is established in this step. This step can be performed as more data becomes available from steps 3 and 4, creating the dataset which would be used in the final step of the method

6. Representation of analysis results

   After assembling all the simulated data, a derived metric suitable for implementation in the HFDM database is required. The model might require time-consuming simulation, but the need to integrate the simulation results into the detection scheme mean that they must be recast in a form suitable for use within an HFDM database.

The following sections detail the pertinent methodology choices which ultimately enable a model-based definition of monitoring threshold and can contribute to the capability of HFDM system to enhance operational safety. The modeling choices are made in accordance with the need to provide additional information for the definition and implementation of key safety metric used in the rotorcraft industry. The choice of these metrics was based on the initial literature survey and the understanding that the primary contributor to rotorcraft accidents is the manner in which the pilots fly the vehicle and those instances when their full control over the vehicle's operation is diminished. LOC-type events make up the majority of accidents, which generates a requirement that the present method can provide results targeted at these conditions. Flight into autorotation, VRS, and rollover on the ground make a large proportion of all LOC events. The goal for this methodology is

therefore to directly benefit the monitoring of this subset of conditions, while considering natural extensions to other operational scenarios.

4.2 Components of a typical nonlinear helicopter model

In creating a physics model from first principles, one of the first tasks is establishing the boundary between the modeled system and the real world[112]. Deciding which effects need modeling and what can be left out is needed to reduce the complexity of the resulting model. Therefore the modeling effort begins with the following research question:

> **Research Question 3:** What physical effects and phenomena must be captured in the helicopter model to make it useful for proactive safety assessments and monitoring purposes?

Based on the insights developed in 2.2.3, a helicopter simulation model similar to the type described by Padfield[94] and Talbot[113] is selected for use in the proposed methodology. A very useful and complete description of the various components that make up such a model is given in Dreier[112], which formed the basis for most of this model's components. The remainder of this section describes the modeling of the relevant model components:

- Fuselage model
- Rotor model
  - Blade dynamics
  - Blade aerodynamics
  - Inflow calculation
- Aerodynamic surface modeling

- Engine/drivetrain representation
- Landing gear model
- Atmosphere and ground reference

The present research is concerned with the type of phenomena that affect the helicopter's performance and handling capabilities throughout the entire flight envelope, but not necessarily beyond. In combination with a review of the most common accidents, a judicious choice can be made as to the type of model which will provide an acceptable compromise between modeling burden and the type of insights which the model will enable. If the desire is to investigate a specific condition, only the effects relevant to that condition need be included. If a more general model is sought, then a wider spectrum of effects must be incorporated. This research makes use of simple models in the initial development effort, and then transitions to a higher-fidelity model that includes the effects of the simpler models in a unified formulation. The remainder of this section details the complete model which should include most of the effects relevant for the initial safety analyses.

4.2.1 Fuselage model

The fuselage is modeled as a rigid body with mass and inertia properties. All forces and moments developed by the rotors, other aerodynamic surfaces and any additional components of the helicopter are applied to the fuselage CG, expressed in the body frame.

The most commonly used earth frame in flight dynamics is the *North East Down* (NED) frame. The NED frame follows the vehicle cg and has the x-axis pointing north,the y-axis is oriented toward east of the vehicle, and the z-axis is pointed down toward the center of the Earth. A common and valid approximation is to assume a flat earth model, which means the curvature and rotation of the earth is neglected. Local variations in terrain can still be included, however.

It is far more convenient to express most of the forces and moments acting on the

fuselage in the body-frame, which has its origin at the vehicle CG and is aligned with the vehicle's primary dimensions. The x-axis points toward the front, the y-axis is oriented toward the right, and the z-axis is perpendicular to both and pointed downwards. The body translational and rotational rates are expressed in this frame. Equations 4.1 and 4.2 are the expressions for the translational and rotational velocities of the vehicle.

$$\boldsymbol{V}^*_{\text{body}} = \begin{bmatrix} U \\ V \\ W \end{bmatrix} \tag{4.1}$$

$$\boldsymbol{\omega}^*_{\text{body}} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{4.2}$$

In the above equations, the * superscript denotes quantities expressed in the body-frame. Fuselage rotational rates are the pitch rate *p*, roll rate *q*, and yaw rate *r*.

To evaluate the transient motion of the helicopter as it reacts to pilot inputs, the derivatives of the translational and angular rates are needed, which yield the translational and angular accelerations as follows:

$$\dot{\boldsymbol{V}}^*_{\text{body}} = \begin{bmatrix} \dot{U} \\ \dot{V} \\ \dot{W} \end{bmatrix} = -\boldsymbol{\omega}^*_{\text{body}} \times \boldsymbol{V}^*_{\text{body}} + (\boldsymbol{F}^*_g + \boldsymbol{F}^*_{cg})/m \tag{4.3}$$

$$\dot{\boldsymbol{\omega}}^*_{\text{body}} = \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \underline{\underline{I}}^{-1}(-\boldsymbol{\omega}^*_{\text{body}} \times \underline{\underline{I}}\boldsymbol{\omega}^*_{\text{body}} + \boldsymbol{M_{cg}}^*) \tag{4.4}$$

where m is the mass of the helicopter and $\underline{\underline{I}}$ is the moment of inertia tensor expressed in the body frame. $\boldsymbol{F}^*_g$ is the force of gravity resolved in the body frame. $\boldsymbol{F}^*_{cg}$ and $\boldsymbol{M_{cg}}^*$ are

the forces and moments generated through the rotor, aerodynamic surfaces, fuselage drag and any other component, such as the landing gear. The above equations are numerically integrated for to yield the body-frame values for translational and rotational rates of the helicopter as shown in eq.4.1 and 4.2. To obtain the vehicle position in the earth frame, with respect to some origin, another transformation is required to bring the body-frame values to the earth NED frame. The transformation of quantities between the earth, body, and any other reference frame used by the model is facilitated through the use of a rotation tensor. The rotation tensor can be constructed using a series of rotations, $\underline{\underline{R}} = \underline{\underline{R}}_{\phi}\underline{\underline{R}}_{\theta}\underline{\underline{R}}_{\psi}$ about the pitch, roll and yaw axes of the vehicle (3-2-1 sequence):

$$\underline{\underline{R}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix} \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.5}$$

The rotation matrix is orthonormal and its inverse and transpose are equivalent. The transpose is used to express the body-frame velocities in the earth frame, yielding the position derivatives:

$$\dot{\boldsymbol{P}}_e = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = \underline{\underline{R}}^T \boldsymbol{V}^*_{\text{body}} \tag{4.6}$$

Calculating of the fuselage orientation in the earth frame is done by transforming the body rotational rates into the earth frame and integrating. The following Euler-angle transformation is used for this purpose, which must be inverted before pre-multiplying the body rotational rates:

$$L_{eb} = \begin{bmatrix} 1 & 0 & -\sin(\theta) \\ 0 & \cos(\phi) & \cos(\theta)\,\sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi)\,\cos(\theta) \end{bmatrix} \tag{4.7}$$

The resulting rate of change of the earth-referenced bank, elevation and heading angles is

$$\boldsymbol{\alpha}_e = L_{eb}^{-1}\boldsymbol{\omega}^*_{\text{body}} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{4.8}$$

The body-frame and earth-frame quantities describing the motion of the vehicle can be assembled in a vector of state derivatives for the fuselage:

$$\dot{\boldsymbol{X}} = \begin{bmatrix} \dot{\boldsymbol{P}}_e \\ \dot{\boldsymbol{V}}^*_{\text{body}} \\ \boldsymbol{\alpha}_e \\ \dot{\boldsymbol{\omega}}^*_{\text{body}} \end{bmatrix} = \begin{bmatrix} \dot{X} & \dot{Y} & \dot{Z} & \dot{U} & \dot{V} & \dot{W} & \dot{\phi} & \dot{\theta} & \dot{\psi} & \dot{p} & \dot{q} & \dot{r} \end{bmatrix}^T \tag{4.9}$$

Integration of $\dot{\boldsymbol{X}}$ is then performed using an appropriate method (commonly a 4th order Runge-Kutta integration scheme) which results in the following state vector:

$$\boldsymbol{X} = [X \; Y \; Z \; U \; V \; W \; \phi \; \theta \; \psi \; p \; q \; r]^T \tag{4.10}$$

4.2.2 Rotor model

The rotor model is the most complex component of the helicopter model and is responsible for the vast majority of the simulated helicopter's response.

If a more detailed analysis of the helicopter is desired, a higher-fidelity approach such as the Blade Element Theory (BET) is needed. The BET allows nonuniform loads to be calculated for blades of arbitrary planform, and the resulting motion to be calculated with regard to the geometric characteristics of the rotor. This allows direct consideration of characteristics which are unique to rotorcraft, such as velocity differences between the advancing and retreating side of the rotor that give rise to asymmetric lift, radial and azimuthal variations in angle of attack of the blades, inflow non-uniformity and the change in lift due to control

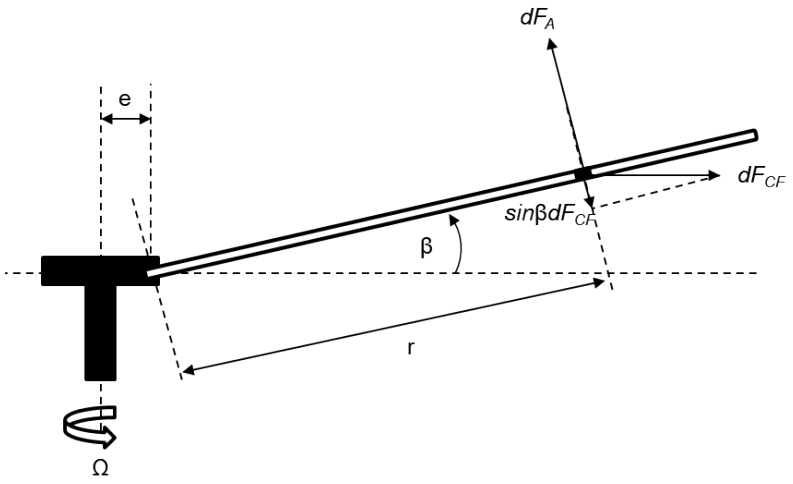


Figure 4.2: Schematic of a flapping blade

inputs. All of these effects give rise to blade *flapping* which is essential for accurate modeling of the helicopter response to control inputs. Figure 4.2 shows a schematic view of a flapping blade, attached with at a hinge offset e. This is representative of an articulated rotor hub, but most rotor types can be represented with the addition of a hinge spring and varying the hinge offset. In this figure, Ω is the rotational speed of the rotor hub, β is the flapping angle, and r is the location of the blade element from the hinge. In a typical blade element application, the aerodynamic and inertial forces at the element are numerically integrated over the blade, which result in forcing terms that cause the blade to flap up or down.

The motion of the individual blade is most conveniently expressed in axis of the blade, so it is important to note which axis is being used. Literature seems to favor one of two frames. Talbot[113] sets the x-axis along the blade, the y-axis toward the the advancing side of a clockwise rotor, and z-axis pointing up when the blade is not flapped. Dreier[112] uses another convention, with x,y,z axes aligned with the body frame when the blade azimuthal station is at $\psi = 0$, which the convention dictates to be straight over the back of the fuselage, pointing at the tail. The selection of blade reference frame has little bearing on the results of the analysis, but the different conventions may be a source of confusion

if not managed appropriately. Another convention that departs from typical definition of angles and angular rates is the definition of the blade flapping direction, which is defined to be positive in the flap-up sense, but due to the orientation of the blade frames, describes a negative rotation about the y axis.

The following equations describe the construction of the blade equation of motion. Evaluating the rotational motion of the blade about the hinge equates the change in angular momentum to the applied moments:

$$\frac{d\boldsymbol{H}}{dt} = \boldsymbol{G} - m_b \boldsymbol{r_b} \times \boldsymbol{a_o} \tag{4.11}$$

where $\boldsymbol{H} = \underline{\underline{I_b}}\boldsymbol{\omega_b}$ is the angular momentum of the blade expressed in the blade frame, $\boldsymbol{a_o}$ is the acceleration of the hinge, $\boldsymbol{r_b}$ the position vector of the blade center of mass,$\underline{\underline{I_b}}$ the blade moment of inertia tensor, and m_b the weight of the blade. The derivative of the angular momentum in the rotating blade frame is

$$\frac{d\boldsymbol{H}}{dt} = \dot{\boldsymbol{H}} + \boldsymbol{\omega_b} \times \boldsymbol{H} \tag{4.12}$$

Substituting the above result into eq 4.11 yields an expression for the blade motion, contained in the $\dot{\boldsymbol{H}}$ term. The $\boldsymbol{G}$ term represents all external moments, $\boldsymbol{G} = [L \ M_A \ N]^T$, applied to the blade. The flapping component of this equation can be expressed in the form, if a stationary hub is assumed:

$$M_A = B\ddot{\beta} + \Omega^2(B\cos\beta + m_b e x_g R^2)\sin\beta \tag{4.13}$$

where M_A is the applied aerodynamic moment in the positive flapping direction, and B is the component of the mass moment if inertia matrix $\underline{\underline{I_b}}$ corresponding to rotations about the y-axis. In reality the hub can translate and rotate which complicates the flapping equation. For flight dynamics simulation and handling-qualities type models, it is common to neglect

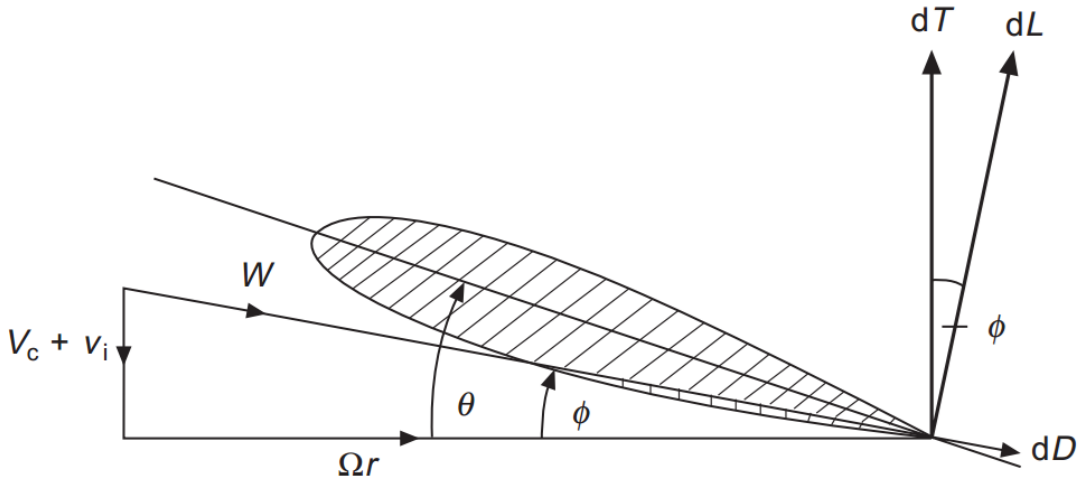


Figure 4.3: Blade section[114]

high-frequency responses of the blade and retain only the first harmonic. This is known as the *quasi-steady assumption*. With the general form of equations (4.11) and (4.12), any hub motion can also be included into the simulation, so the present approach will use an appropriate level of simplification for the conditions at which the simulation will be executed. For example, the blade angular velocity, in the blade frame, when considering body pitch rates and a constant rotor rotational speed can be shown to be (using Dreier's convention and for a clockwise rotor):

$$\boldsymbol{\omega_b}^* = \begin{bmatrix} p\,\cos(\beta)\,\cos(\psi) + \sin(\beta)\,(\Omega + r) + q\,\cos(\beta)\,\sin(\psi) \\ -\dot{\beta} + q\,\cos(\psi) - p\,\sin(\psi) \\ \cos(\beta)\,(\Omega + r) - q\,\sin(\beta)\,\sin(\psi) - p\,\cos(\psi)\,\sin(\beta) \end{bmatrix} \tag{4.14}$$

This expression is inserted into (4.12) and solved, in this case for the flapping component. With appropriate simplifications, such as the quasi steady assumption, small flapping angles, and constant rotor rotational speed, it is possible to derive analytical expressions of the flapping equation. This periodic equation can be split into constant, $\sin\psi$ and $\cos\psi$ components, which is a form frequently used in flight dynamics applications[114].

The next component that must be evaluated is the aerodynamic moment M_A, which

can be determined by calculating the lift on each blade segment and summing the moment about the hinge due to each element. A section of the blade is shown in 4.3. The blade section sees the incoming airflow due to the blade around the hub, Ωr and the component of vehicle velocity in the direction of the blade section. The pitch of the blade is set by a mechanical linkage that varies the pitch of the blade as it goes around the azimuth, but the instantaneous angle of attack is different due to the wind velocity in the vertical direction. The climb velocity V_C and the induced inflow velocity v_i serve to reduce the blade angle of attack by angle ϕ, called the inflow angle. Thus the blade section sees an angle of attack $\alpha = \theta - \phi$. The sectional lift and drag, dL and dD are evaluated with the familiar equations:

$$dL = \frac{1}{2}\rho W^2 \, c \, dr \tag{4.15}$$

$$dD = \frac{1}{2}\rho W^2 C_d \, c \, dr \tag{4.16}$$

The moment M_A is obtained by resolving the thrust and drag forces in the direction of the flapping motion, and apply the resultant force with a moment arm equal to the radial distance.

$$dT = \cos\phi dL - \sin\phi dD \tag{4.17}$$

$$dQ = r\left(\cos\phi dD + \sin\phi dL\right) \tag{4.18}$$

In a numerical scheme, this is performed for each blade section and the sum of the individual contributions gives M_A. At this point, corrections can be applied to account for root cutout and tip losses[91, 92]. A similar calculation in the planar direction is used to obtain the rotor torque, which must be balanced by the tail rotor and is related to the rotor power by the relationship $P_{rotor} = \Omega Q$, where Q is the rotor torque. The forces, moments, and torque/power for the tail rotor are obtained in a similar fashion, though frequently flapping

motions are neglected to reduce the computational burden.

4.2.3 Inflow

One of the most important aspects of rotor modeling is calculating the airloads, which depend primarily on the inflow through the rotor disk, or else the airflow experienced by the rotor. There are many choices available in the helicopter literature [115, 116], with varying degree of fidelity and ease of implementation. For the purpose of this effort, the inflow model should consider flight into the VRS regime, since this condition is a known safety hazard. Several model have been proposed that consider VRS in particular, such as the Wolkovitch [117], Peters[118], ONERA[119] and Johnson [120] models. Most of these models provide an expression for the boundary of the VRS region based on a physically relevant criterion, such as vortex convection speeds, stability, V_z drop and similar. This type of boundary is ideal for immediate online application for monitoring or limit protection[119]. Several of these models also provide an extended inflow model that is valid up to and within the VRS region. Further, Johnson shows a good agreement for dynamic simulation of entry into VRS, and for this reason, this model was implemented in the current simulation and used in both as a standalone simplified condition boundary and within the dynamic model.

$$\nu = \frac{\nu_h^2}{\sqrt{V_x^2 + (V_z + \nu)^2}} \tag{4.19}$$

$$\nu = a + bV_z + cV_z^2 + dV_z^3 \tag{4.20}$$

In Eq.4.20 a is usually taken to be zero, which allows the exact fitting to two points and a derivative at one of the points. ONERA performs a similar procedure for their inflow model, with the exception that the a term is retained. Equation 4.20 is valid for a constant forward velocity, which is the reason it only contains V_z terms. To include the fit in the flight dynamics simulation and in the VRS boundary, a non-dimensional lookup table is

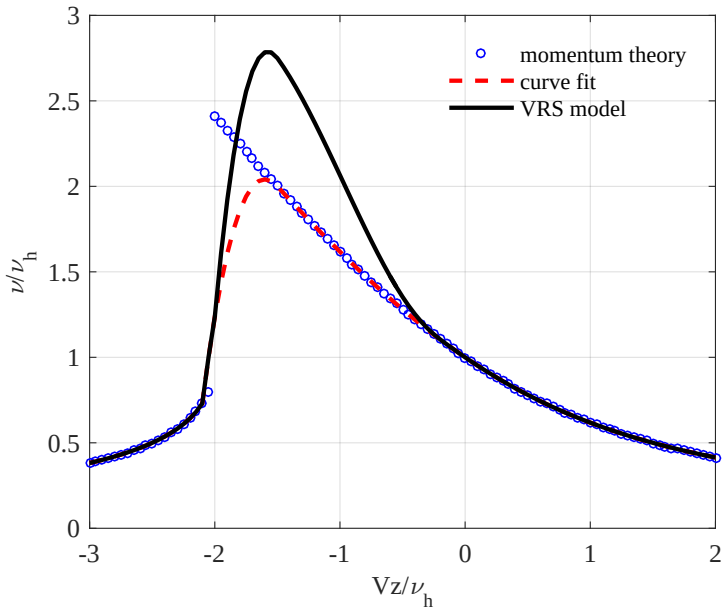


Figure 4.4: Inflow model extension to include the VRS regime [120].

created, with the values multiplied by the hover induced velocity to obtain dimensional quantities. A further development shown in [120] is the extension of the VRS model to time-domain simulation by casting the result as a first order differential equation witht the time constant defined in equation 4.21.

$$\tau_{rev} = \frac{0.7}{\lambda_h} = \frac{0.7}{\sqrt{\frac{C_T}{2}}} \tag{4.21}$$

Since the same approach is taken for both main rotor and tail rotor inflow, two additional states are added to the simulation and integrated in time.

The above calculation is relatively straightforward, but depends on an expression for the inflow v_i. This has been the focus of much research and there is a continuous effort to improve the prediction of inflow across the rotor. The simplest form is the one used in Eq. (2.1), derived for hover and under the assumption of uniform distribution across the rotor. This momentum-theory model can be extended for forward flight, which is the form

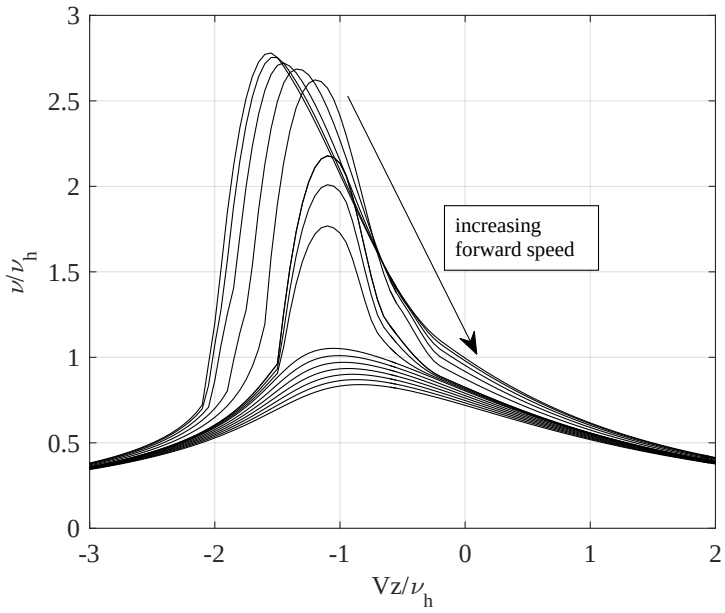


Figure 4.5: Inflow model in forward flight.

required in Eq. (2.2). The normalized equation for inflow in forward flight is:

$$\lambda = \mu \tan \alpha + \frac{C_T}{2\sqrt{\mu^2 + \lambda^2}} \tag{4.22}$$

As can be seen from this equation, it is a recursive equation with the inflow depending on the denominator and varying with forward flight speed, captured in the advance ratio μ. A Newton-Raphson iteration is frequently used to calculate this equation within a few steps, though numerical instabilities are possible at low thrust settings. This form of the equation has been used for many flight dynamic analyses, but is known to be a gross simplification of the flow through the rotor, especially at conditions away from hover[91]. For example, in descending flight the equation breaks down and is no longer valid when the helicopter enters vortex ring state (VRS). Thus for maneuvering flight, a more accurate representation is usually needed and the present model will require such a representation to allow more accurate analysis of descending flight conditions such as autorotation.

Improvements to the standard uniform inflow model are possible by introducing az-

imuthal and radial variation, so that the inflow expression takes the form $v = v(r, \psi) = v_0 + v_c r \cos\psi + v_s r \sin\psi$. If the model is to account for the effects of descending flight and transient maneuvering imparted by the interaction of the rotor wake, then wake models or dynamic inflow models is required[121], such as Pitt-Peters and Peters-He inflow model[118, 122]. These models use the concept of apparent mass of the volume of air affected by the rotor to derive a differential equation that can be used to calculate the transient inflow velocity as the vehicle maneuvers.

$$[M] \begin{bmatrix} \dot{v_0} \\ \dot{v_c} \\ \dot{v_s} \end{bmatrix} + [L]^{-1} \begin{bmatrix} v_0 \\ v_c \\ v_s \end{bmatrix} = \begin{bmatrix} C_T \\ C_{M_Y} \\ C_{M_X} \end{bmatrix}_{aero} \tag{4.23}$$

where v_0, v_c and v_s are the uniform, lateral, and longitudinal inflow. C_T, C_{M_Y} and C_{M_X} are the coefficient of thrust and the aerodynamic moments at the rotor (neglecting inertial hub forces). $[M]$ and $[L]$ are the apparent mass and gain matrices which are derived from vortex theory. A model of this type can be included with the rest of the vehicle states and be integrated within the same routine to provide a relationship between the blade loads, rotor flapping response, and inflow. Another possible approach to account for the inflow variation with flight condition is the Ring Vortex Model[123]. This model represents the wake of the rotor with discrete rings whose effect on the flow is calculated using the Biot-Savart law. By manipulating the relative position of the rings to account for forward or descending flight, good agreement with experiments can be achieved.

The final output of the rotor model are the forces and moments which act at the hub. There are three force components, namely the thrust, H-force and Y-force, and three moments that represent the pitching and rolling moment and torque. The application of rotor forces away from the fuselage cg creates additional moments, as does the reaction at the blade hinges when in rotors with hinge offset or in hingeless rotors.

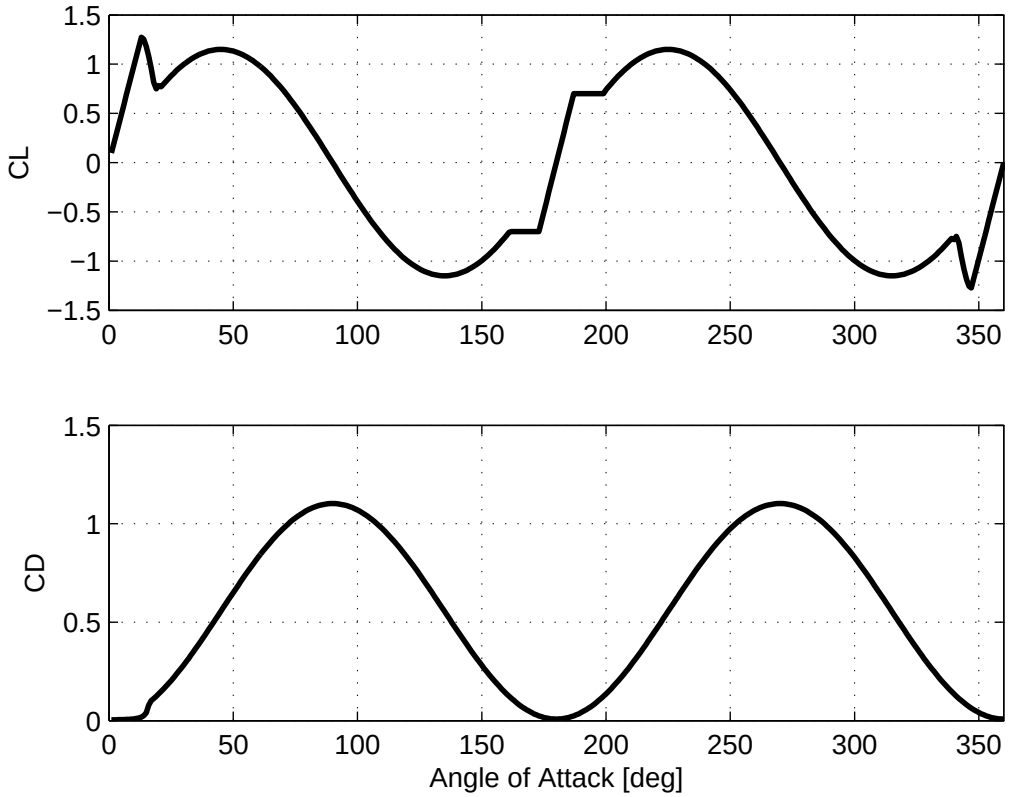


Figure 4.6: C_L and C_D variation with angle of attack[92]

4.2.4 Aerodynamic surfaces

In addition to the aerodynamic forces generated by the blades, helicopters typically have vertical and horizontal aerodynamic surfaces at the tail. The forces on these surfaces can be defined if the overall wind speed is reoriented in expressed in the local frame of reference, aligned to the aerodynamic surface. Then the aerodynamic angle of attack can be obtained and the lift, drag and moment coefficients can be estimated. For both the blades and the aerodynamic surfaces, the model will use a 360-degree approximation to the steady lift and drag curves as developed by [92].

4.2.5 Landing gear

In addition to the typical flying responses, a helicopter simulation with a safety perspective should also capture the ground-based behavior of the helicopter. However, in most dynamic helicopter simulations, the landing gear are either entirely omitted or receive minimal attention[91, 94, 113, 124]. Landing gear are understandably left out from most flight simulation analyses, as the behavior of the aircraft while on the ground is rarely of interest. However, since many helicopter accidents occur on the ground as a consequence of dynamic or static rollover [81, 80], it is important that a vehicle model used for safety assessments contains a representative landing gear model.

Most of the landing gear models for aircraft simulation have been developed to assess the longitudinal dynamics of fixed-wing aircraft for the purpose of evaluating their landing performance [125, 126, 127]. In these cases, only the vertical response of the landing gear compression and the friction due to the tires and brakes are needed, with the friction being the dominating effect [128]. For helicopters, the longitudinal response of the landing gear is important, but is a relatively small effect when compared to the lateral effects. Whereas a stationary fixed-wing aircraft has little opportunity to generate lateral forces, a stationary helicopter can indeed generate large lateral forces and rolling moments, leading to potentially dangerous conditions. For this purpose, several researchers have investigated landing gear models and the ground-based dynamic response of rotorcraft, especially with regard to the ship-helicopter interface.

One of the early investigations in this area is the one by Blackwell and Feik, who conducted a very relevant study of on-deck helicopter dynamics using a simple fuselage and landing gear model [129]. Their model consisted of a rigid fuselage with oleo struts represented by nonlinear springs and dampers, and a spring/damper tire model. This model facilitated an investigation into the response of a helicopter on a moving helideck. Another set of studies has been performed by Langlois and colleagues[130, 131]. Their model is depicted in figure 4.7, and consists of three tires attached to nonlinear oleo struts, along

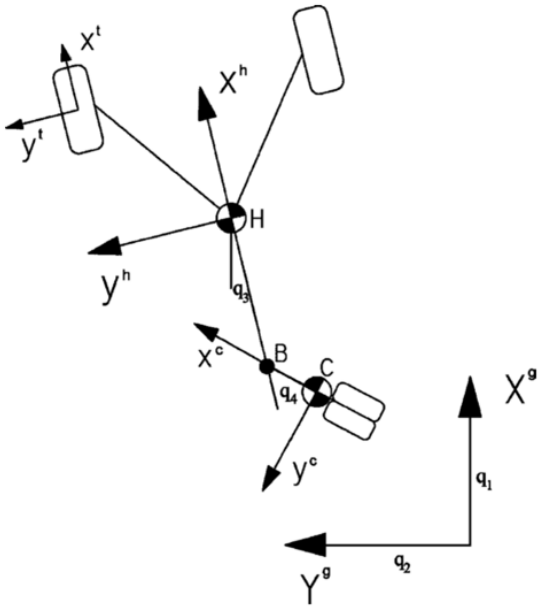


Figure 4.7: On-deck helicopter model[131]

with a castering nosewheel. They considered lateral forces and obtained good correlation with test data. A more recent analysis was conducted by Leveille[132], who used a similar approach to investigate the ground handling and securing of a helicopter on a ship deck.

Both Leveille[132] and Linn and Langlois[131] used the tire models contained in the report by Smiley and Horne[133]. The Smiley and Horne report is a seminal work concerning tire dynamics for aircraft, and includes many test results along with simplified empirical models of tire behavior[133]. This type of empirical model gives the lateral and longitudinal force due to the tire, for various tire inflation pressures, temperatures, ground friction properties and other parameters. Another popular empirical model is the Pacejka series of *Magic Formula* tire models, which can contain upwards of 70 model parameters that account for a wide range of operating condition variations[134, 135]. However, helicopters might not be a good candidate for such empirical representations for two reasons. First, the concept of the ”relaxation length” seems problematic, as is the undefined value of the tire sideslip when the tire is stationary. The relaxation length means that the tire must roll a certain distance for the lateral deflection of the tire to return to the resting value. In

reality, it seems as though the vehicle/fuselage would be forced to acquire a position such that the tire lateral deflection returns to zero, rather than the tire remaining deflected when no additional forcing is applied to the fuselage. The second aspect which may limit the applicability of existing empirical formulas is the representation of tire lateral force in terms of tire sideslip. It is believed that this representation is due to the nature of the experiments conducted to evaluate tire behavior, but the effect is that at low speeds or at rest, the existing empirical models break down and are not suited for safety assessments using dynamic models at these conditions.

Of the non-empirical tire models, the brush and string models stand out as potential candidates[136, 137]. However, as in the case of the empirical models, the increased number of parameters may unnecessarily complicate any future model calibration efforts, with a comparatively small payoff in terms of model accuracy. In addition to their ability to represent longitudinal and lateral tire dynamics down to the resting condition, these types of models are well suited for investigations of high-frequency and high-speed tire phenomena, which are very important for vehicles that derive most of their dynamic response from the tires. For helicopters in ground operations, the forces and moments generated by the tires are primarily in the form of forces applied at the contact point, and any moments about the strut itself can be neglected due to their small size.

For this investigation, the model by Blackwell and Feik[129] was chosen as a starting point. They used a simple representation of the vehicle with three landing gear struts, and used discrete deflections of a in-plane spring at the ground contact point to represent tire deformations - which are ultimately equivalent to the force applied to the strut. In the current implementation, a small modification was made to improve the ”rolling” characteristics of the Blackwell model, specifically in the form of the lateral deflection of the ”tire” that is related to the immediate orientation of the strut x-axis in the ground plane. By allowing the tire contact patch to travel in the instantaneous body x-direction, while the fuselage is undergoing arbitrary motion, a deflection is allowed to develop, which results in a force

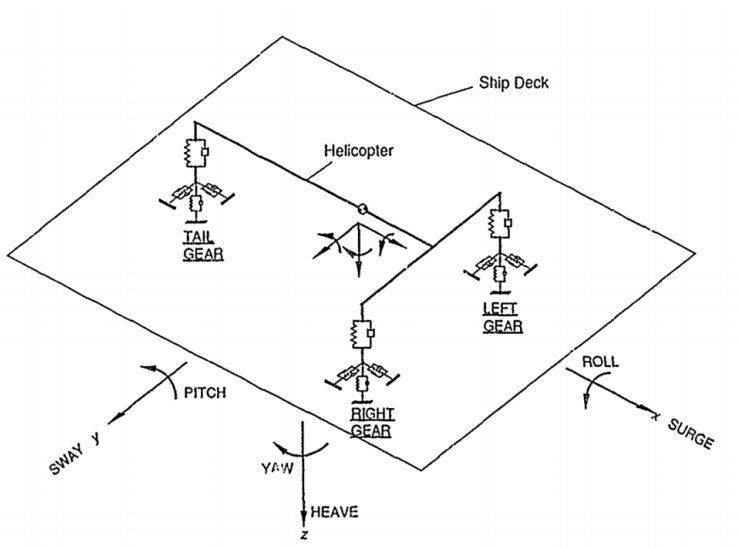


Figure 4.8: On-deck helicopter model by Blackwell and Feik[129]

being applied to the strut. With this modification, the landing gear is able to develop cornering forces as a result of the vehicle yaw angle with respect to the velocity vector, as well as generate lateral forces when the vehicle is stationary. The maximum force that can be developed in this manner depends on the friction coefficient between the tire and the ground, as well as the normal force at the contact point. If exceeded, the tire is allowed to slide, while maintaining the maximum deflection and force.

4.2.6 Environment

The environment model is used to set the global conditions where the model will be operated. The primary elements of an environment model are representations of the atmosphere and the ground. A comprehensive simulation should include some provision to represent variations in wind and potentially turbulence, such as the Dryden model. The ground can be similarly represented using a global model where every coordinate position is associated with an elevation. The ground model can be further enhanced with gradient information for the local ground plane, surface characteristics and obstacle locations. The use of these ”environment” models can also be thought of as a placeholder for future extensions of the

method, such as the inclusion of input from databases that store ground obstacle information, helipad locations and descriptions, and databases of hazards such as power lines. The consideration of such potential data sources will allow the method to operate on real-world information, if such information becomes available during the term of the present investigation.

4.2.7 Summary of helicopter dynamic modeling

The proposed method depends on the availability of a helicopter model that contains the appropriate effects and is valid at the conditions of interest. The effects and model components described above should provide such a model, though much of the capability can be obtained using far simpler models. The following hypothesis can be formulated:

> **Hypothesis to RQ3:** An appropriately detailed physics based model can be used to identify potential accident conditions, without the need for direct observation, if the physics responsible for the accident mechanism are included and well represented in the model.

The challenge implicit in the above hypothesis is that a model will never fully represent the real system and at all possible conditions, and the only truth model is the system itself. However, for the conditions of interest, it can be reasonably assumed that the physical effects are known and can be included in the model. The consequence of this hypothesis being justified is that accident conditions identified using the model are representative of the real helicopter's safe boundary.

In an effort to manage development risk, and with a view toward established comprehensive codes for helicopter analysis, the modeling will proceed in a two phase progression. In the first phase, a low-fidelity model will be developed that will be used within a narrowly-defined operational scope. This model should also facilitate the development and evaluation of the other methodological components. The second phase should see the

implementation of a high-fidelity model within the method in order to extend the analysis to additional flight conditions and capture any relevant effects that had been omitted in the simpler model. The method itself is being developed with the consideration of allowing other models to be used within the general framework, so that the appropriate model for a given flight condition under investigation can be used. The next section discusses distinction between online and offline uses based on the evaluation time of the math model (roughly corresponding to static and dynamic models), but any model capable of taking in inputs in the form of a state vector and generating either a derived parameter or a range of related trajectories would be applicable within the DEMMoS method.

4.3 Selection and propagation of analysis scenarios

Online use of models for monitoring

In the case when a mathematical model can be evaluated in a single pass or with minimal iteration, the inputs to the model can be taken directly from the flight data in the monitoring database. At present, the intended use of these faster models is in an *off-board* monitoring system, so that the entire model evaluation and condition detection is performed on the ground, post flight, without the more stringent requirements of real-time evaluations. In a real-time environment, there are usually known hardware and software limitations that allow a certain number of model iterations per unit time, with fewer iterations possible when the model is of higher complexity and requires longer evaluation time. For off-board and post-flight data analysis, there is no requirement for real-time calculation, but quick model execution is still preferred given the potentially large amounts of data being processed. In this context, **online monitoring** refers to the fact that these model-derived metrics are calculated at the time of evaluation of the flight data concurrently with any other regular events which are monitored in the HFDM database. These models make some key assumptions regarding the vehicle and its operation that enable their application:

1. Quasi steady-state operation
2. Nominal rotor RPM
3. Primarily longitudinal motions
4. Control inputs are as required to maintain the instantaneous flight condition

This approach is based on the notion that a model-derived parameter can be more physically relevant to a particular flight condition than the raw data, especially in cases when the available data do not include control, engine, or air data parameters. Aside from the calculation of the derived parameter, the operation of the model-based safety events is identical to that of existing HFDM events. The ability to implement model-derived monitoring parameters in an identical fashion to existing HFDM events is the primary criterion that determines their applicability in an online sense. If a model evaluation requires significantly more iterations than a comparable HFDM event, then it would not be suitable for online use and an alternative is required. One such alternative is to run an extensive set of simulations separately from the flight data monitoring database, and pre-compute derived metrics which can be evaluated quickly within the constraints of the database, which is discussed next.

Offline simulation for monitoring indicator development

The **offline** portion of the method is to pre-calculate model derived parameters in advance of their use within the online detection scheme using a dynamic model of the helicopter in flight. This approach can be thought of as a generalization of the condition-specific monitoring using static models since a more complete model of the vehicle is used, which can be evaluated at a variety of flight conditions, potentially uncovering combinations of flight states, control inputs and vehicle configuration parameters which are not immediately known to be related to an accident. In this manner, the physics contained in the model yield the indication of unsafe flight states, and this result is then used to relate the flight state to

the risk encountered given a particular control input. Starting at some initial condition, the model will have to be manipulated in some way to elicit behavior that results in either a safe or an unsafe outcome. The manner in which this is accomplished has a bearing on the general applicability of the method. Therefore the following research question is formulated:

> **Research Question 4:** Starting from a relevant initial state and vehicle configuration, how should the model be controlled throughout the simulation?

It is important to note that the use of simulation can only identify adverse conditions that arise out of the modeled physics. The **true limits** limits of the vehicle might arise from complex interactions of various subsystems, and a detailed model which contains these effects could provide the means of determining situations when critical subsystem states manifest them as operational limitations. At present, the primary interest is in the gross motions of the vehicle, so the limits encountered in simulation stem from dynamic constraints such as contact with the ground. At the same time, any of the modeled parameters could represent a limit if they reach a physically impossible state (eg. flapping beyond flap stops, exceeding structural strength), so the capability of the approach would only increase when more detailed models are used.

4.3.1 Prescribed maneuvers and scenarios

In the following, several alternatives are presented which pertain to the methodological requirement to operate the model during simulation and allow conclusions to be made regarding the safety and risk of the operation.

One possible way to fulfill this requirement is to prescribe both scenarios and trajectories over the course of a simulation run. A well-known source of prescribed maneuvers for helicopters is the ADS-33E-PRF document, which describes maneuver sequences known as Mission Task Elements (MTEs)[138, 139]. This type of analysis is frequently performed

during design or handling qualities assessments, since the prescribed maneuvers are used as standards against which different designs are compared, or shown to satisfy design requirements. Often the desired maneuver is not specified directly, but is expressed in terms of a cost function that is used in an optimal control framework. Bottasso used this approach in analyzing ADS-33 maneuvers, as well as in evaluating critical trajectories[102, 140] using a dynamic model and an optimal control function for each maneuver. Several other optimal-control approaches have been demonstrated in literature, especially in the study of the helicopter H-V diagram[99, 100, 101, 141].

Evaluation of longer flight segments can also be accomplished by arranging pre-defined maneuvers in sequence. The AIRSAFE concept, created by Burdun et al.[142] used a customized structure to represent complex accident scenarios, spanning a portion of the flight deemed relevant to the accident. For example, a takeoff accident would include a set of segments with discrete transitions, propagated through to an accident outcome. Rearranging and perturbing the segments was used to investigate the ”neighborhood” of the pre-defined accident scenario. For this analysis, Burdun et al. used real accident data to develop each scenario, identified key decision points during the flight and assigned probabilities to alternative actions.

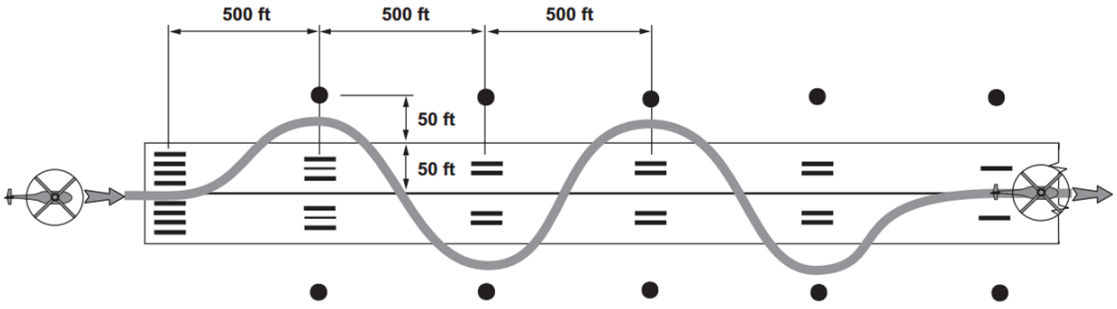


Figure 4.9: Suggested layout of slalom course[138]

4.3.2 Exploratory analysis of flight conditions

A more exploratory approach would be to set a starting point only, but allow the vehicle to be maneuvered arbitrarily during the simulation. This type of propagation of the controls during simulation is attractive for establishing the envelope around a trim point or point recorded in the data. Ansari[143] used an approach where dynamic trim was first achieved at various points in the flight envelope, and then the simulated aircraft was commanded to reach a different trim point. The boundary between achievable trim points and flight states where the vehicle could not trim was established as the envelope. Figure 4.10 shows a result from this exploration of the flight envelope using the dynamic trim concept.

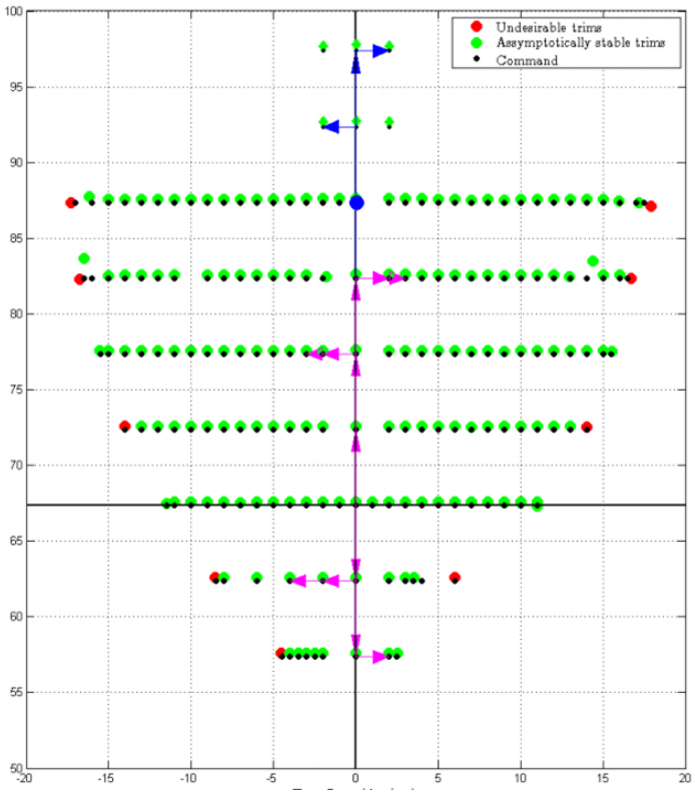


Figure 4.10: Envelope determination[143]

The techniques discussed so far assume a starting point, and either prescribe a starting point and a trajectory or a trim point and a control input/new trim. For the type of safety assessment planned in this research, the second approach seems like a good candidate for

exploring the envelope around conditions with known proximity to failure because a wide range of possible trajectories are investigated by manipulating the inputs.

Monte Carlo approaches

A potential criticism for an approach using control input and parameter sweeps might be that in reality the inputs are not likely to remain constant. There are several domains where the evolution of a dynamical time-dependent system is of concern, with a special interest in eliciting a broad range of outcomes. This type of "exploratory" approach is well suited to the present application. In the field of nuclear powerplant engineering and telecommunications, researchers often deal with highly reliable systems whose simulation requires either a large number of evaluations or a special technique to efficiently provoke the behaviors which are relevant from a safety perspective. These techniques frequently employ a Monte Carlo approach as the basis for their development. The additional components are almost always aimed at reducing the computational burden while steering the simulation towards types of operation where failures can be identified.

The steering of a system toward unknown path evolutions to a new and potentially unobserved state is similar to path planning, where the system is not propagated through the physical space but throughout the simulated domain. One of these approaches is the particle filter (PF)[144, 145].

Studies indicate that in the limit, the PF approximates the Kalman Filter solution, which is known to be optimal for linear systems with Gaussian noise characteristics. One of the benefits of using a particle filter is the ability to capture arbitrary distributions of the outcome. The particle filter framework is a potential candidate to allow control inputs variability to be taken into account with the use of random system inputs. In addition, the particle filtering framework introduces the concept of importance sampling, which represents the distribution of the output through the position and weight of the particles. A similar approach is the particleRRT[146] (rapidly exploring random tree). Both of these methods

have been explored extensively in literature from path planning to the simulation of rare events in highly reliable systems.

At the root of the particle filter simulation approach is stochastic propagation of a system model to a future state using multiple concurrently evolving trajectories, with the simulation time horizon divided into intervals or segments. Each trajectory ("particle") represents one possible way the system can evolve from the start state at the beginning of a simulation segment to the end state at the end of the same segment. At the end of any given segment, the distribution of end states describes the set of possible outcomes, given the input variables and their ranges. In a filtering set up, the segments are set up to coincide with the arrival of new measurements, which are combined with the simulated trajectories using Importance sampling techniques. Another approach used in both path planning and rare event simulation is Importance splitting. A particularly interesting technique is that suggested by Pradlwarter[147], where a dynamical system is steered toward regions in the response domain that of particular interest, as defined via an importance function which determines the trajectories to be split.

Most particle filtering techniques are defined for discrete-time systems and use some recursive formulation to propagate the state. These models are frequently formulated as variations of Hidden Markov Models or similar state transition machines, and researchers have noted the challenges of applying the probabilistic framework to a continuous system or a system with a large domain of possible trajectories[148]. In the present case, the equations of motion are integrated to approximate continuous states, so modification to the original algorithm would have to be made[149]. The alternative is to discretize the simulation runs into smaller segments that create the effect of a discrete-time system, which is compatible with the iterative nature of algorithms such as the particle filter. If no measurements are available, the particles can still be propagated to establish a pseudo-distribution of outputs.

This brief review suggests that several approaches may be applied in a simulation sce-

nario for the purpose of eliciting safety hazards. Overall, the Monte Carlo approach is of particular interest as it reduces the need to define pre-conceived targets or assumed forms of the vehicle trajectory during simulations for safety analysis. To answer the question at the beginning of this section several candidate alternatives are proposed in the remainder of this section and are later assessed in section 5.2.

4.3.3 Simulation propagation in DEMMoS

This subsection discusses several approaches used to initialize and propagate the simulation through time in a manner that would be useful for safety analyses. The control inputs are of crucial importance to making any conclusions about the future states of the vehicle, except in extreme conditions where any control input would result in a crash or other failure. The goal of the appropriate input strategy is two-fold: to elicit the types of behavior which result in increased risk, and to enable monitoring for the identified risk by mapping the information generated during simulation to the pilot's inputs and any other relevant parameter.

Discrete sweep

The first approach is to determine an intial condition and select a control input which is applied at a particular instant in time, maintaining the same command throughout the remainder of the simulation. This approach has the advantage of being conceptually simple, and allows a direct mapping between the outcome and the inputs, as was done in [150] using the 2-D tipover model described in section 5.3.1. During a discrete sweep of control inputs, the simulation is repeatedly executed using a constant control input and the response of the vehicle is tracked. Given a set of limit parameters, the time history of the vehicle response can be evaluated for any excursions beyond the specified threshold. The discrete sweep is analogous to the DOE approach in computational experiments, where a range of simulation inputs are used. DOE techniques for efficient sampling of the input space exist,

but in a nonlinear simulation this may cause important effects to be missed. Therefore the approach taken here is to simulate all combinations of input parameters, given a finite set of discretizations of each input variable's range. Another important note is that in many cases, the results of a simulation study must be mapped to the input space, which in most cases is the set of control inputs that are used to operate the vehicle. If the control inputs are trajectories, as opposed to fixed values, then this mapping is complicated and a straightforward relationship does not exist. For this reason, the discrete sweep approach is preferable if the goal is the development of criteria for monitoring adverse conditions, expressed in terms of the variables defining the input space. The mapping of simulation results to a fixed value in the input space is also used in some more online limit protection systems, where the assumption is that the limit would be repeatedly computed at consecutive instants in time, thereby mitigating the concerns of mapping dynamic trajectories to a fixed value in the input space.

Other approaches

The simple sweep analysis would often represent the worst-case scenario, since in many cases it is the extremal control inputs which bring the vehicle to a critical flight state in the shortest amount of time[150, 151, 107]. Over longer prediction horizons, the simulation may be repeated using a initial condition chosen from some intermediate set of results. In this fashion, trajectories are generated that originate at a common starting point, and evolve in response to the entire admissible range of control inputs (quantized into a finite number of levels). Each of these trajectories can then produce additional branches which evolve independently as a result of the control inputs applied to the system. The ability to perform a simulation of this sort requires the simulated system to approximate a Markov process, which means that at any time step the system state depends only on the previous step and not on the entire time history up to that instant in time. The dynamical systems and simulation integration schemes used in this work have this property, enabling the time domain

of interest to be split into smaller segments, and to assign different control inputs to the individual trajectories during each segment. Each trajectory can be considered a separate ”particle”, even with the overlap during the initial segments. This approach is termed ”sequential sweep” and is investigated as an alternative in the experimentation section. Yet another approach is to use fully randomized inputs, which creates a large number of independently evolving trajectories over the prediction horizon. This ”randomized input” approach to simulating a dynamical system should in the limit visit every possible path and capture the full variability of potential system responses. The following hypothesis is stated and evaluated through experimentation in a later section:

Hypothesis to RQ4: Starting with the same initial condition as the baseline scenario propagation, a particle-type approach using randomized inputs can identify the outcome derived using the baseline approach.

The consequence of this hypothesis being confirmed is that the method can identify hazards without explicit specification of the maneuvers to be performed. Further, by approximating variable pilot inputs, a more realistic representation over longer prediction horizons can be achieved than is achieved by the steady control input method.

4.4 Representation of simulation outputs

In the case of condition-specific and static models, the evaluation of incoming flight data would be instantaneous based on flight data alone in an online detection scheme. In order to eliminate some of the assumptions of the simpler condition-specific models, a more detailed model may be used, which is frequently accomplished in an offline setting. In the case of dynamic simulation using nonlinear models, the simulation produces a sequence of simulated vehicle states, both relating to the motion of the vehicle and any additional simulation parameters which are available. Section 4.2 described the components that would

make up a nonlinear helicopter model with the appropriate fidelity for application within DEMMoS. A comparatively long evaluation time and a need to map the simulation outputs to the flight condition of interest necessitate the ability to propagate the simulation through time, as described in section 4.3.

The literature reports various metrics that have a relationship to safety, including control margin, limit margin, energy based and time based metrics. Of these, the first-hitting-time metric was selected for its general applicability and high-level view of safety. There does not seem to be a straightforward way to formulate and test a hypothesis regarding this choice, but the present experience suggests that the metric will provide adequate distinction between safe and unsafe conditions. Additionally, a time-based threshold could be specified by the operator to establish acceptable margins of safety which can be related to the separate limits on individual parameters. Another reason the time based metric is deemed adequate is the prevalence of similar representations of safety and reliability in other fields.

In particular, the temporal aspect of the output allows one to determine the instant when a limit parameter is exceeded and relate that result to an earlier instant. This aspect of the offline approach allows the use of a first-hitting-time metric, which is the defined as the first sample from the limit parameter that has exceeded a pre-specified value. Once a limit is observed in simulation, and the time to critical conditions is determined, the outputs would have to be summarized in a manner that allows implementation in a monitoring database. In addition to the time-based metric, it is expected that the simulations will generate large amounts of information which will have to be efficiently integrated in a monitoring environment in order to make the results from the simulation experiments useful for actual safety monitoring schemes.

One option is to use the results directly to help identify traditional HFDM events, though it is desirable to reduce the burden on the analyst in defining safety events. One potential approach to implementing these results in practice would be to generate thresholds on parameters such as control inputs, based on a particular value of the time to a

critical condition. This type of application would result in a static boundary, but could be useful for the analysts defining the detection boundaries and as training material.

A second approach would be to provide the data directly and compare against these data using interpolation. There are a variety of interpolation algorithms available, but the practical concern is that interpolating a high-dimensional lookup table for every data point might be problematic in a real-world HFDM application given the intent to continue populating the repository of simulated results over time.

The alternative is to provide a regression model which has been fitted to the data and can be rapidly evaluated at any point where the regression is valid. There are a number of techniques which perform regression on data of various complexity, so it is reasonable to hypothesize that a technique that extracts this knowledge and represents it in a functional form can be identified and potentially used for implementation of identified safety limits. Therefore an approach to represent this relationship using some form of a surrogate or supervised learning technique is needed, which leads to the following question:

Research Question 5: How can the time-history output of the simulation be interpreted in a general framework and be implemented as a monitoring boundary?

Specifically, the results of interest are the time to reach a critical value of any limit parameter given a flight state as specified by the incoming data stream. This functionality of speeding up the analysis of incoming flight data and providing a representation of simulation-based metrics can be achieved using a number of statistical learning or surrogate modeling techniques, though the nature of the response would dictate the appropriate choice.

The response surface methodology (RSM) is one common choice when generating surrogate models as a result of a large number of computer experiments[152]. The most frequently used approach to generate the relationship between input variables and outcomes is multi-dimensional linear regression. However, as will be seen during this work,

the response of dynamic experiments using helicopter models is not linear, and contains additional complications which make the application of linear model less straightforward.

Since the multi-dimensional surface of the simulation response can contain highly non-linear regions, truncation, and otherwise be of a type that is not conducive to linear representations. One way to achieve an improved regression is to transform the original data such that it becomes linear in the transformed domain. Given the unknown and potentially complicated form of the response, the transformation approach could require a large number of operations before the data could be used, without any indication whether new data would require yet another transformation.

Yet another approach is to separate the original domain into smaller sub-domains where the response is approximately linear. Qiu investigated methods to fit regression models to discontinuous surfaces[153] using a three stage approach of separating the domain at the discontinuity locations and fitting separate models in each sub-domain.

In addition to these options, there is an extensive body of literature of potential alternatives, especially if supervised learning techniques are considered. There an innumerable variations and customization of these methods and algorithms, and a full investigation is beyond the scope of this work. Section 2.1 regarding statistical approaches in flight data monitoring mentioned several techniques which have been used in aviation, and in particular, neural networks stand out as a very flexible and powerful learning approach[62, 63]. There are numerous applications of neural networks in related fields, and there is a renewed interest in their applications in part due to recent demonstrations of very promising results in the area of deep learning[60, 154]. This leads to the following hypothesis:

Hypothesis to RQ 5: An appropriate neural network technique will enable accurate and efficient representation of the outputs suitable for evaluation in large databases, while preserving all relevant characteristics of the simulation analysis results.

If this hypothesized model is successful, the result will be a detection boundary that

will not suffer from the type of simplification that typical "safety events" require before implementation is practical. This component of the methodology aims to bring the results of a complicated analysis closer to the operator and enable actionable detections of flight hazards. In addition, in order to provide flexibility to the operator in deciding where this threshold is set, the entire set of output data should be made available to the operator/HFDM system.

4.5 Methodology summary

The methodology described in this chapter is expected to provide an overall capability which has not been reported in HFDM use, and has the potential to help reduce risk through detection and mitigation. The methodology relies primarily on a modeling and simulation component which can be evaluated to yield enhanced and predictive measures of the risk associated with a particular type of operation. These metrics are to be implemented in an online fashion within the HFDM system as an additional set of events to the ones already in place. Such model-derived events can be defined for a variety of conditions for which mathematical formulations exist or can be developed. The method also provides for the use of more complicated and computationally expensive models through the use of a structured simulation and analysis set of components which take a model through a range of possible trajectories in order to identify the presence of unsafe flight states. It is envisioned that the simulation portion of the method would be evaluated off-line and generate a number of simulated trajectories ahead of the use of the results for monitoring. The simulations would expose hazards within a region around the initial condition used in the simulation , and would be repeated at other conditions in a continuous fashion while populating a repository of simulated flight data. DEMMoS is intended to combine the results of many such simulation analyses in order to enable online monitoring using a supervised learning technique, enabling the detection of safety hazards that stem from the physical mechanisms which have been captured in the model. It is important to note that the structure of the

method was intended to be model-agnostic, as long as the model is an accurate representation of the physical system that is the vehicle, and can be evaluated using the proposed scheme.

Figure 4.11 shows the expected use of the DEMMoS method as an addition to existing HFDM system, with the primary result being the improved detection of adverse flight conditions through the definition of model-based safety events. Such events have the potential to be defined accurately and early in the operation of an HFDM system, increasing the opportunities to detect and mitigate unwanted trends in helicopter operations. The following chapter provides support to the methodology choices and shows the application of DEMMoS on several relevant flight conditions, with the goal of enabling and improving the detection of safety hazards in helicopter operation. A final demonstration is also performed to showcase an how the method is to be initialized and operated to the point of generating a detection metric for a particular flight condition.

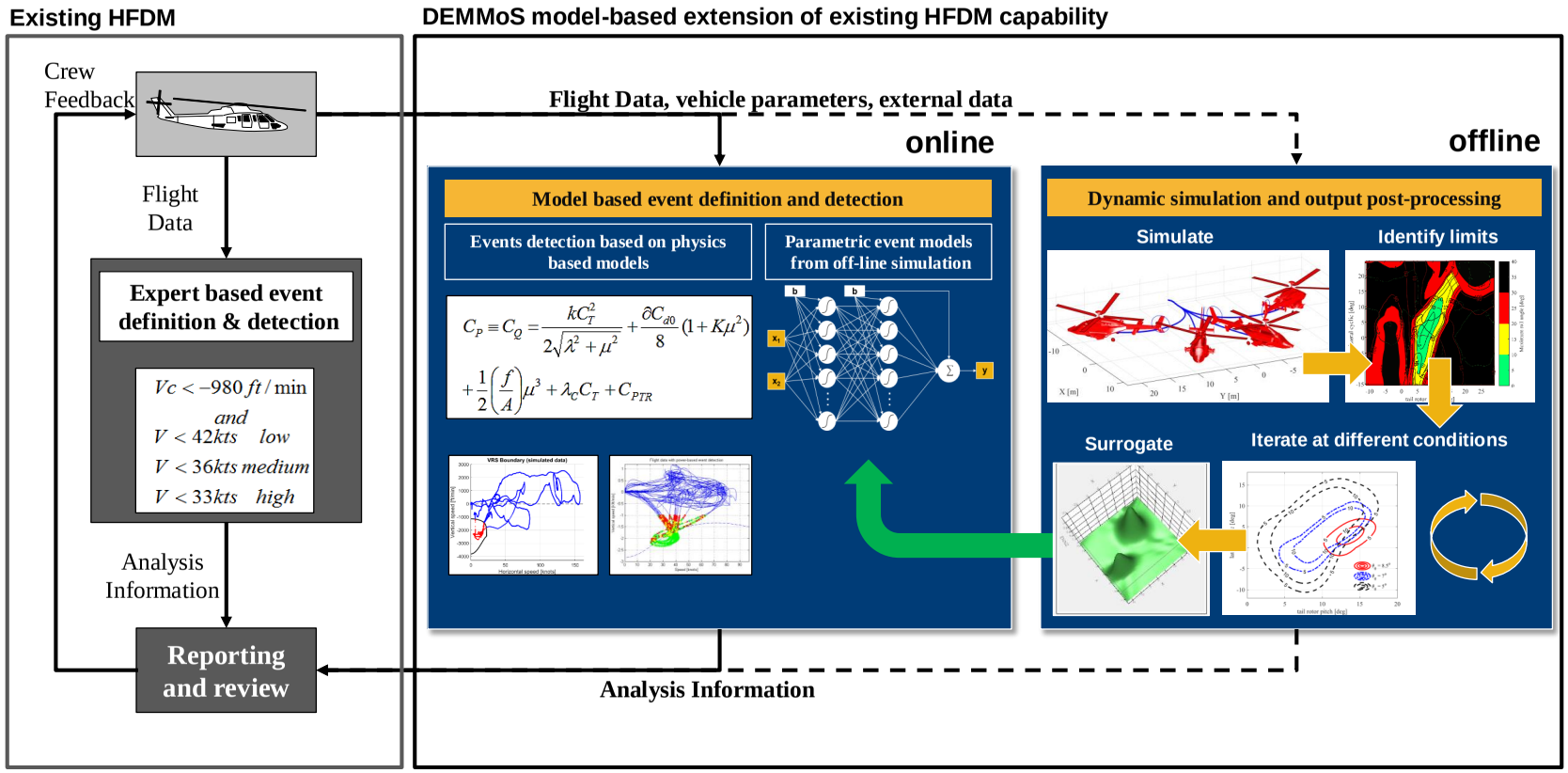


Figure 4.11: DEMMoS implementation

CHAPTER 5
EXPERIMENTATION AND RESULTS

This chapter describes the experiments performed in support of the development of the model-based methodology and the definition of safety metrics based on physical principles. The first part of this chapter pertains to condition-specific models which have been shown to improve HFDM detection ability. The approach is extended to a more general dynamic modeling approach in the later sections.

5.1 Condition-specific models

5.1.1 Vortex Ring State Model use in HFDM

In HFDM implementations using simplified models for condition detection, the model is used in the same manner as regular safety events, with the exception that the monitoring levels (condition indicators) are physically meaningful quantities output by the model instead of pre-defined thresholds established by experts and fine-tuned over a period of time.

This principle was applied to the detection of flight near the Vortex Ring State regime. The VRS boundary is usually expressed in terms of the horizontal and vertical velocities V_x and V_z [120], and many researchers are in general agreement on the location of this region. Figure 5.1 shows this boundary for a simulated flight data record, defined based the inflow model described above. The boundary is defined as the point of vertical descent rate where the inflow gradient changes, seen as a "bump" in fig.4.4. The helicopter was intentionally flown into the VRS region several times, and at each time step the flight state of the vehicle was checked against the boundary valid at that flight state. In this case the detection is based on a logical test against the VRS boundary, which is identical to the approach broadly used in HFDM. The primary difference is that the VRS boundary

shown here is scaled with the hover induced velocity, so vehicle size and weight as well as atmospheric conditions are taken into account. In Figure 5.1, the boundary shown is drawn for the nominal flight condition, and several points where VRS is indicated appear just outside of this nominal boundary. If the boundary is re-drawn for the instantaneous flight condition at the particular flight data point, then all detected points would be within the boundary, as expected. Figure 5.2 shows the altitude profile of the same simulated flight over time, with ground height shown. The data were generated using the X-Plane® flight simulator for the purpose of evaluating the operation of the VRS detection, since this type of occurrence has not been observed in the data available during this work. This type of graph is more commonly found in HFDM, and it clearly shows the portions of the flight where VRS was experienced. In this flight, VRS was deliberately approached and entered 3 times, with a resulting increase in vertical descent rate. The recoveries was performed using both the traditional procedure and the newly popularized "Vuichard" technique, which uses tail rotor thrust to help exit the influence of the vortex.

As an example of how this result may be used in typical HFDM, the flight's trajectory is shown on a Google Earth map (fig.5.3), which is one the ways operators analyze flights and assess any hazards encountered during the flight. The trajectory is a clear indication of the rapid and nearly vertical descent during vortex ring state encounter. Though this example uses a simulated set of flight data, the analysis is nearly identical in the real-world case. The primary difference would be related to issues with noise and other artifacts in the data, though most of the hardware used for this purpose is very reliable and performs sufficient conditioning of the data before the data upload is completed.

Because the detected condition is related to the properties of the flow through the rotor, the same principles can be applied to any rotorcraft, conceivably allowing the same model to be applied across a range of vehicles with minor modifications. This is a direct benefit for HFDM operation, since the manual safety event definition can be aided by physically meaningful values for boundary parameters. A further benefit is that the boundary shown

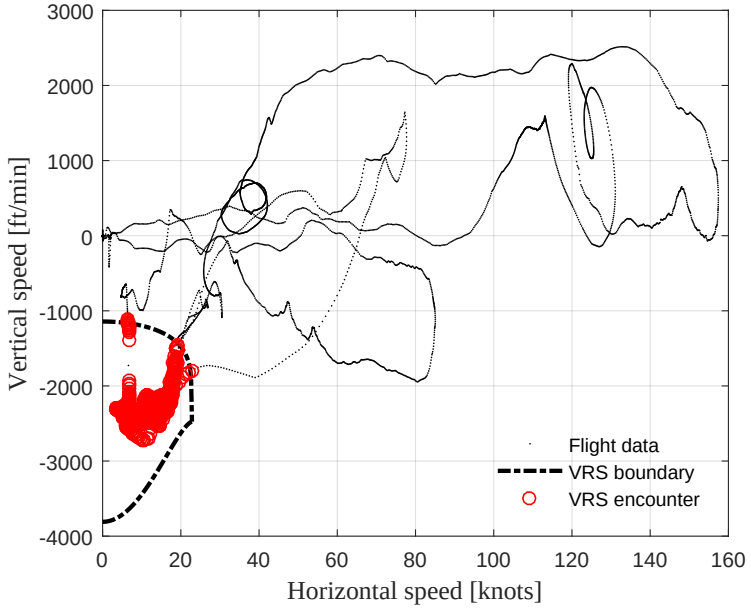


Figure 5.1: Detection of VRS using boundary from Johnson[120].

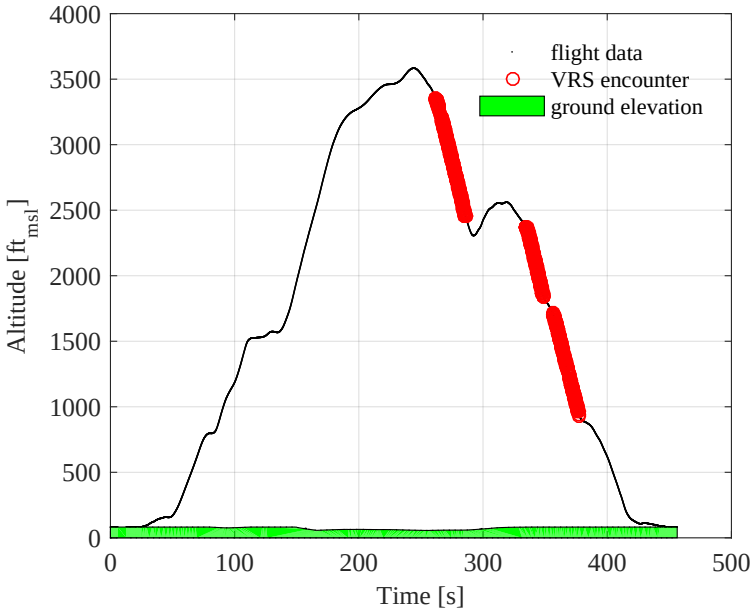


Figure 5.2: Flight profile of simulated flight with VRS encounter.

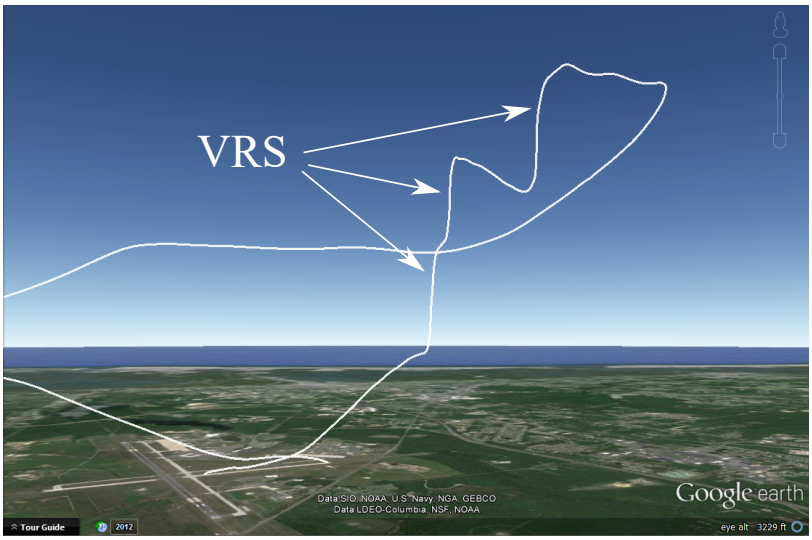


Figure 5.3: Flight track of simulated flight with VRS encounters.

in Figure 5.2 is not static, but changes according to the vehicle configuration, as well as the instantaneous flight conditions such as load factor. In this limited case, all three occurrences of VRS were detected, which points to the feasibility of using the parametric VRS boundary for this type of detection in an on-line HFDM system.

5.1.2 Autorotation monitoring

During the course of this study, flight data recorded by a helicopter operator became available. Most of the flight data were collected during uneventful routine flights, but a single flight was identified where large vertical speed deviations had been detected. It was determined that this flight represented practice autorotations. This was confirmed by the fact that the flight data contained associated safety event definitions used by the operator to assess the aircrews performance during training. The purpose of monitoring autorotation practice is to detect any instances when the trainee pilots perform deviate from standard procedures and place the helicopter at an increased risk of low RPM and consequently increase the risk of a hard landing or crash. A further point of interest was the indication that in this particular instance the events have been tagged as nuisance alerts, since the meta-data for

the flight data record contained a note from the analysts regarding pilot comments and the fact that the detected events had been triggered unnecessarily. This is a prime example of situations where the analyst would either modify the event and associated detection limit or disregard the flight data record.

Specifically, the limits set by the operator were crisp logical rules defined on flight parameters relevant in autorotation. A threshold vertical speed of 1000 feet per minute was assumed, and three values for horizontal velocity V_h were specified to correspond to different severity levels.

$$V_h < 40\text{kts}(\text{low}) \tag{5.1}$$

$$V_h < 35\text{kts}(\text{medium}) \tag{5.2}$$

$$V_h < 30\text{kts}(\text{high}) \tag{5.3}$$

In [91], a practical approach to estimating the descent velocity in autorotation is presented where the power requirements of the vehicle are equated to the descent velocity through the exchange of potential energy for the energy consumed by the rotor and expended on overcoming air drag. When the exchange of potential energy is sufficient to cover the power required by the vehicle at a particular flight speed, the calculated power required becomes negative. This point of transition is the descent speed in autorotation, and this boundary can be defined for a particular vehicle and a range of forward velocities.

In Figure 5.4, the flight data from the autorotation training flight is shown in terms of forward speed (GPS measurement) and vertical speed. The limits in Equations 5.1-5.3 are shown through the color-coding of the flight data points which lie below the 1000ft threshold. The points progressively to the left of the figure are more "severe" because the rotor requires more power at low forward velocities than at intermediate velocities close to the point of minimum power.

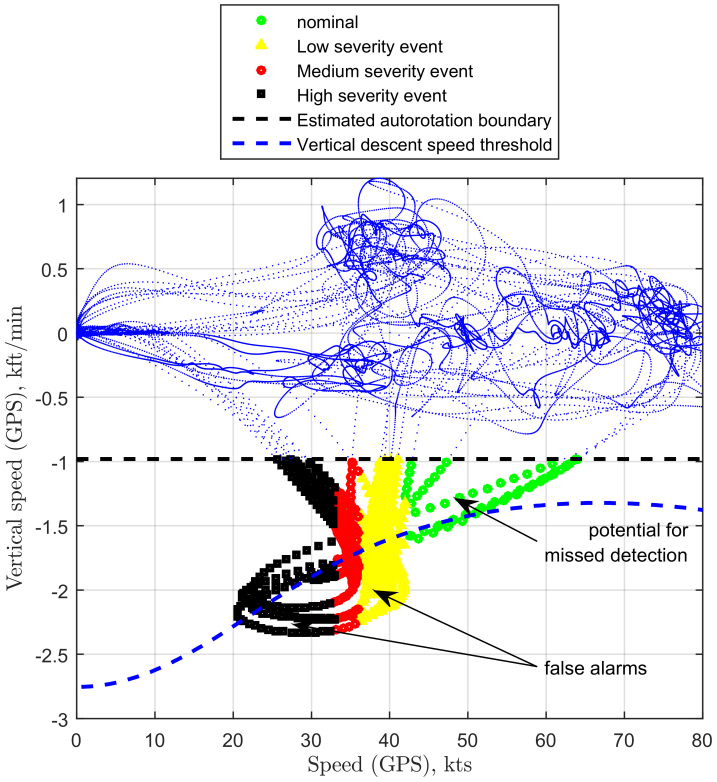


Figure 5.4: Monitoring of autorotation practice

A model-based analysis of the same data set was performed using the expression for power required in forward flight in 5.4, expressed in terms of ideal descent velocity given the instantaneous flight conditions.

$$C_P = C_Q = \frac{kC_T^2}{2\sqrt{\lambda^2 + \mu^2}} + \frac{\sigma C d_0}{8}(1 + K\mu^2) + \frac{1}{2}\frac{f}{A}\mu^3 + \lambda_c C_T + C_{P_{TR}} \tag{5.4}$$

The detection thresholds in this case were related to the power requirement, with low severity events being triggered at -5% power. In autorotation, the calculated power should be slightly negative to account for tail rotor, transmission, and any other losses. This estimate can certainly be refined, but the results in Figure 5.5 show an improved correlation between the ideal autorotation boundary and the flight data points marked as low or high severity autorotation events. Only two detection thresholds were used due to the arbitrary nature of the 5% level, and any flight data points with a power more negative than 5% were deemed acceptable.

The results show that it may be possible to establish monitoring events using even a simple model of a performance condition. While the 1000 fpm threshold was retained, inside the prescribed autorotation region the model-based method shows improved correlation to the detection boundary. Additional analysis on this particular data record revealed that significant atmospheric winds were present during the flight, which was partly responsible for the false alarms.

However, after accounting for the wind component, it was shown that the original thresholds would still yield false alarms, whereas the model-based approach shown here in conjunction with the wind estimate eliminated false alarms of data points that are clearly within the autorotation regime. These results are highly encouraging an point to the benefit of using physics-based models in HFDM. It is important to note that this is a limited test case and further flight data records would be required for additional validation. At present,

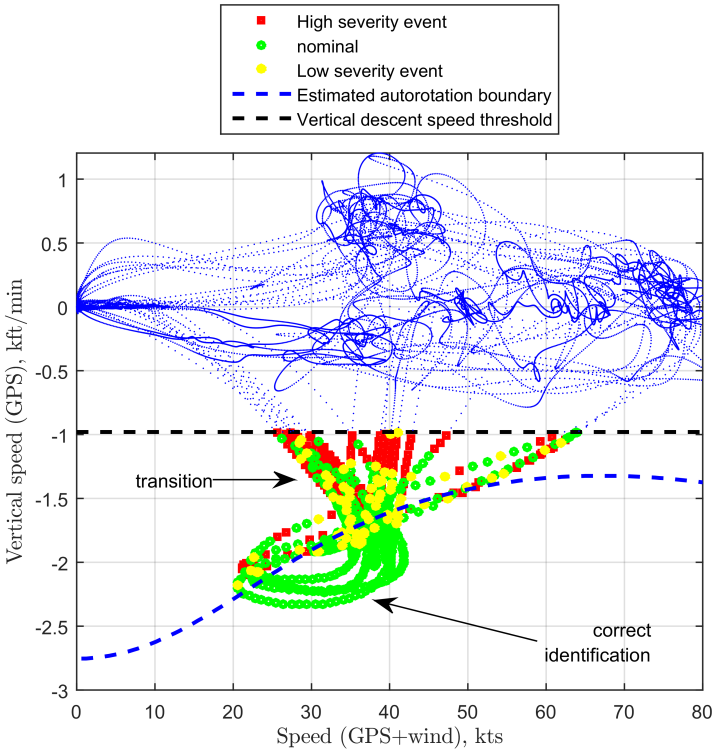


Figure 5.5: Model-based monitoring of autorotation practice

this model has been validated as effective by subject matter experts in the helicopter domain [84].

The results shown in the two cases of VRS detection and autorotation monitoring confirm the potential for the model-based approach to provide benefits over the current state of the art in Helicopter Flight Data Monitoring. Providing operators with an effective means to detect hazardous flight conditions though model-based detection thresholds could reduce the false alarm rates and missed detection, as demonstrated by the improved agreement between the detection boundary and the autorotation boundary. This result supports the hypothesis to Research Question 3, which is that an appropriately detailed model can adequately identify hazardous conditions. These two models were defined explicitly for the conditions where they are applied, so the second part of the hypothesis requires further support. To move away from condition-specific models, are more general formulation is needed in the form of dynamic models, which are evaluated next.

5.2 Analysis of simulation input strategies

This section deals with the evaluation of several approaches for their applicability in DEMMoS as a means to manipulate the simulation in a way that uncovers potential hazards. In the review of applicable techniques, Monte Carlo approaches in various particle-based forms were identified, and two variations are compared in this set of experiments. The "truth model" is an optimal control solution for a simple dynamical system described in the next subsection. The goal is to determine an approach that can balance the needs to explore the operational space while enabling monitoring of conditions that have been identified during this exploration, and whether the basic discrete-sweep approach can achieve similar results.

5.2.1 Spring-mass-damper model

One potential criticism of the discrete sweep approach may be that it doesn't represent the real operation of the vehicle over long prediction horizons. This is a reasonable concern, as pilots are not likely to maintain a fixed control input over an extended period of time. An optimal framework[107, 106, 100], a realistic pilot model[155, 156], or both[102], can be a much closer approximation to the expected manner of operating the vehicle in reality. In order to evaluate different approaches based on the open-loop control input scheme used in this work, a 1-D linear spring-mass-damper example from [107] is implemented. The damping coefficient is $\zeta = 0.7$ and the frequency $\omega_n = 2.0$ rad/sec, with the following form when expressed in state-space:

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -4.0 & -2.8 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \tag{5.5}$$

The calculation of an optimal input control time history depends on the choice of cost function, which is minimized with respect to the control input. In the present case, the cost function is constructed in a way that penalizes time and excessive control effort. Many other functions may be applicable, including ones that contain the constraint to reach $X = 5m$, which is the limit parameter critical value. In the present case and in [107], the cost function in Equation 5.6 is used. The W parameter is a weight assigned to the control inputs. In the case when W is zero, the maximal control input is given (in this case the upper bound was set to 30 m/s^2), and with progressively higher weight the control input is penalized so that the optimal solution resembles that in Figure 5.6.

$$J = \int_{t_0}^{t_f} (1 + 0.5 W u_p^2) \mathrm{d}t \tag{5.6}$$

For the purpose of calculating the integral in Equation 5.6, the control input time series was discretized at fixed intervals, resulting in segments of constant control input. The

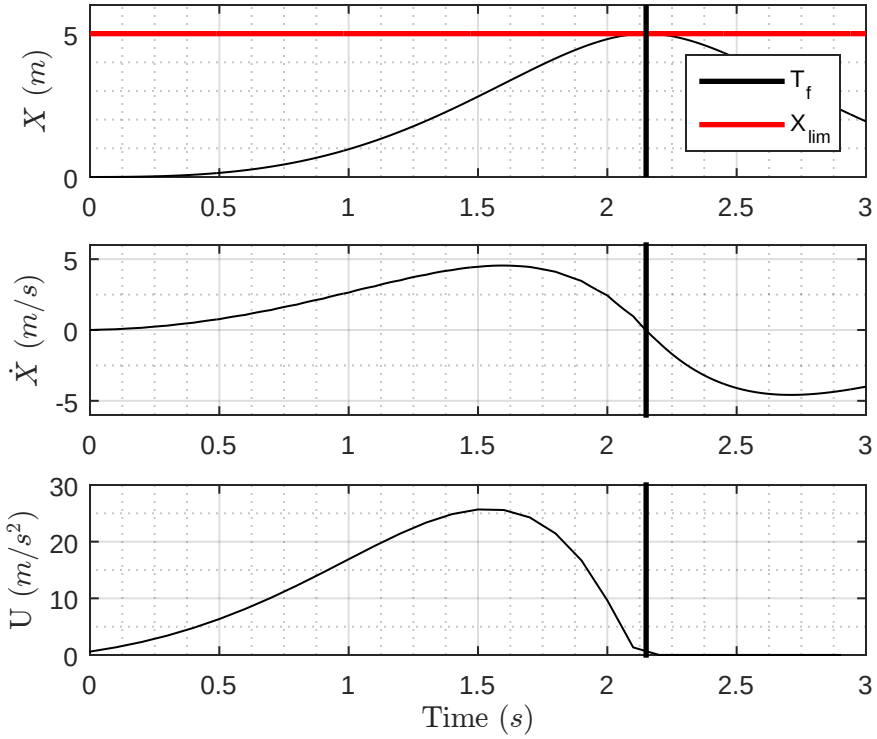


Figure 5.6: Optimal control solution for 1-D example

control values were weighted according to the segment duration and summed. The value of the control input during each segment thus becomes a design variable for constrained nonlinear optimization problem. Figure 5.7 shows the convergence performance of several individual optimizations of the spring-mass-damper control problem. As the number of control segments is increased, the optimal solution is rapidly approached and increasing the number of segments has little additional benefit. The conclusion is that as long as the number of segments allow the resulting (stepped) control trajectory to approximate the form of the true optimal control input, it will be possible for the optimization to converge to optimal solution. In the case of the spring-mass-damper, the solution does not improve significantly beyond 10-15 segments.

Once an optimal control is computed, it is common to map this control time history to the initial condition using a fixed surrogate value [107], and re-compute the solution at later instants in time. It is this notion of repeated evaluation that allows the use of fixed input

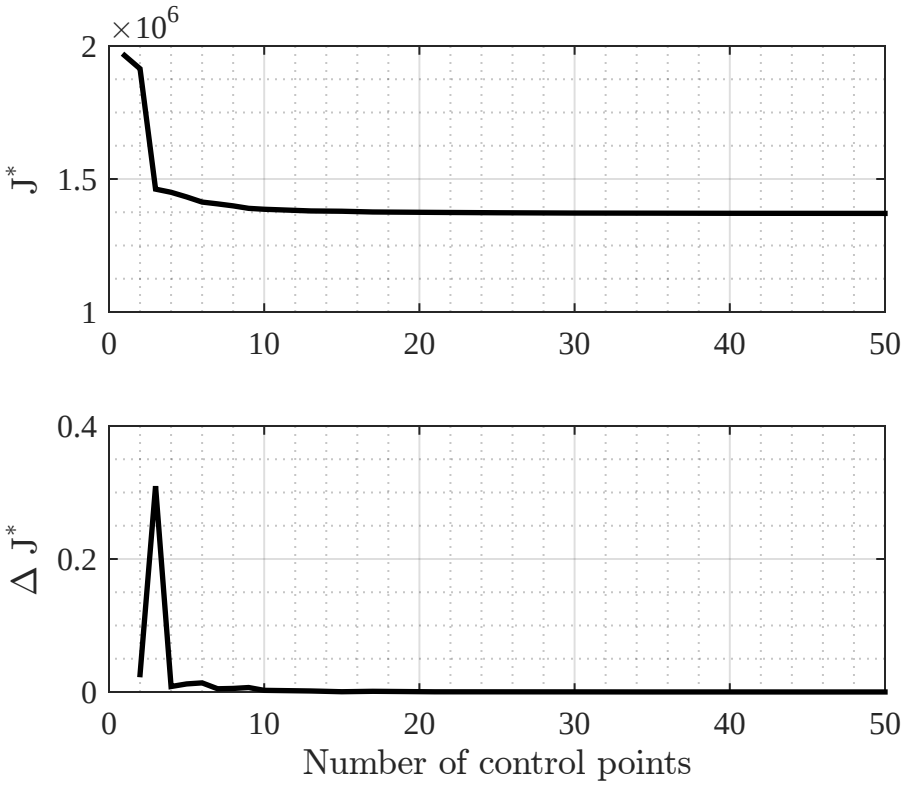


Figure 5.7: Convergence of optimal solution for range of collocation points

instead of the optimal solution, by changing the value of the fixed input at each intermediate condition where the solution is recomputed. The same approach can be applied without the optimization aspect, but through the use of the DOE approach in a sequential manner. Below is the evaluation of two candidate techniques to extend the discrete sweep using fixed inputs to a more exploratory approach.

5.2.2 Sequential sweep of control inputs

The essence of the sequential sweep approach is to allow the simulation to investigate as many paths as possible, which is true in the limit. Allowing every realization of the simulation to "go anywhere" and from there to proceed to "everywhere else" in the state space should in principle visit every possible trajectory realization. While this is not practical in the sense that such an investigation would require unlimited computational resources, it is interesting to compare this approach against the optimal solution. Unlike the opti-

mal solution, the sequential sweep does not assume any cost function or form of control input, though the number of quantization levels (the values the control input can be set to) and the number of segments or discretizations of the time horizon have to be determined. This is similar to the discretization of a continuous time domain in some of the approaches reviewed in Section 4.3. Figure 5.8 shows one such simulation of the spring-mass-damper system described by Equation 5.5, where the simulation was performed in two stages. During the first stage, trajectories were generated for each of the three control input quantization levels. During the second segment, starting at $t = 1.5$, each trajectory from the first segment was allowed to propagate in 3 more instances, one for each control input level. In the cases where the control input remained the same as that which was used during the previous segment, the trajectories follow the same path as those generated in the discrete sweep case. It is interesting to note that if the limit parameter $X_{lim} = 5$ is the only metric of interest, the steady inputs reach the limit fastest. In addition to these trajectories, the solution contains trajectories where the control input is changed to one of the other possible values. In terms of the output X, these additional input cause every possible path to be explored. This first example is primarily used to demonstrate the concept of the sequential sweep, but it is clear from Figure 5.8 that none of the trajectories are particularly close to the optimal path shown in Figure 5.6. The optimal path, shown in red, was generated by a time-varying control input which results in a path with more variation than can be reproduced with only two temporal segments with constant inputs.

$$D = \frac{\sum_{i=1}^{n}(X^i - X^i_{\text{opt}})^2}{\sqrt{n}} \tag{5.7}$$

The sequential sweep is in essence a sequence of Monte Carlo-type analyses that propagate the simulation throughout a wide range of states. While a solution of a particular form is not sought out expressly, the sequential application of all possible input allow the simulation to approximate many different types of scenarios which would otherwise have to be defined explicitly. In Figure 5.11, the distance calculated in Equation 5.7 is plotted

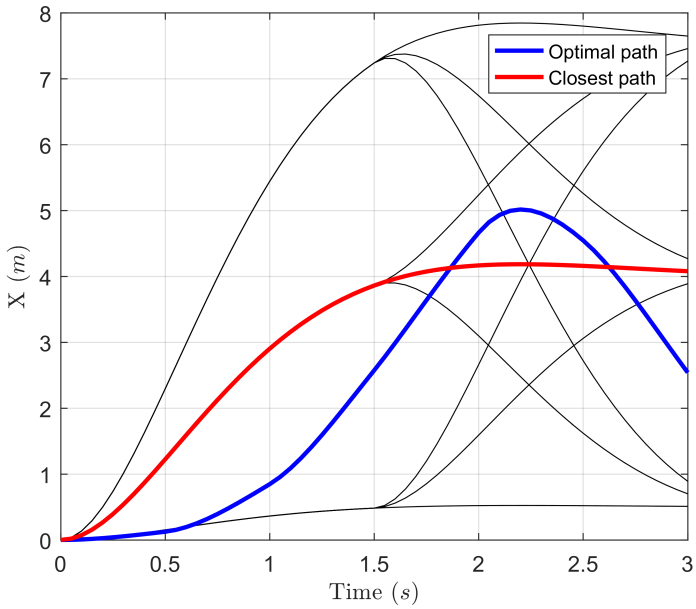


Figure 5.8: Sequential sweep (3 levels, 2 segments)

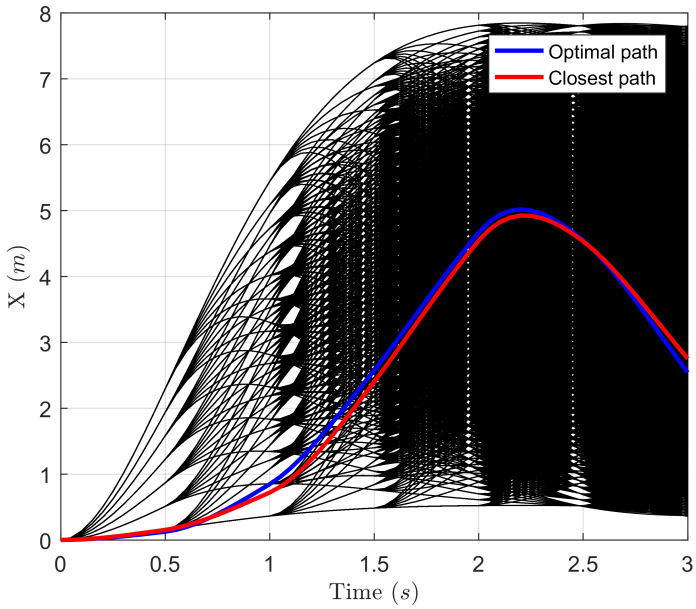
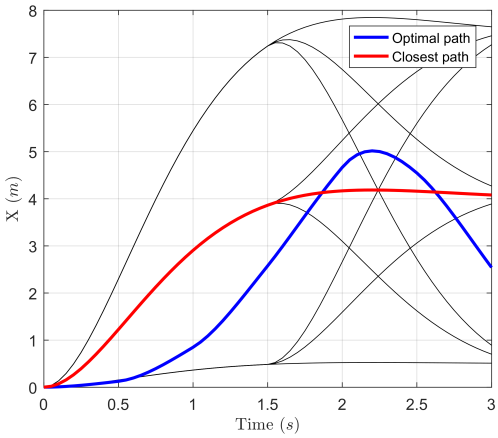
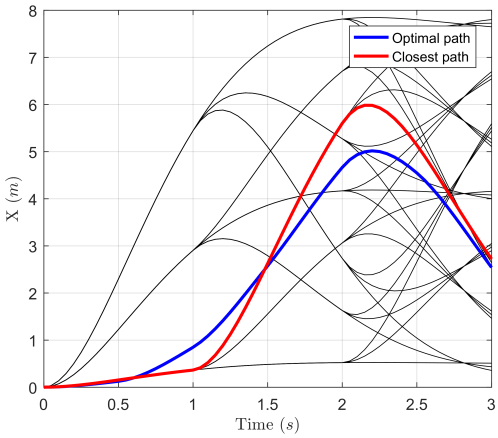


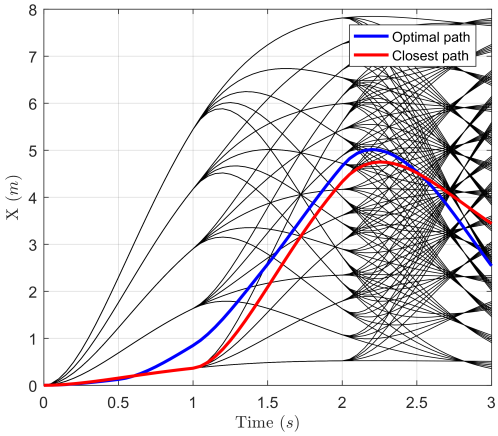
Figure 5.9: Sequential sweep (7 levels, 6 segments)



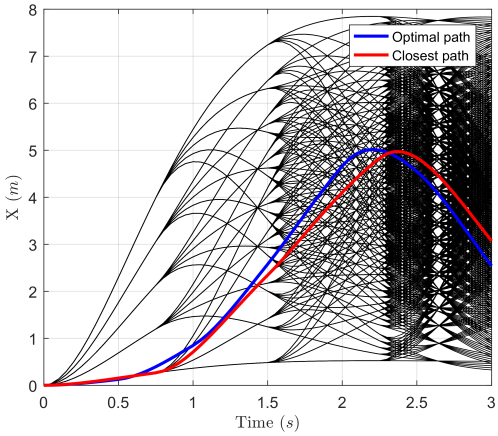
(a) 2 segment, 3 levels



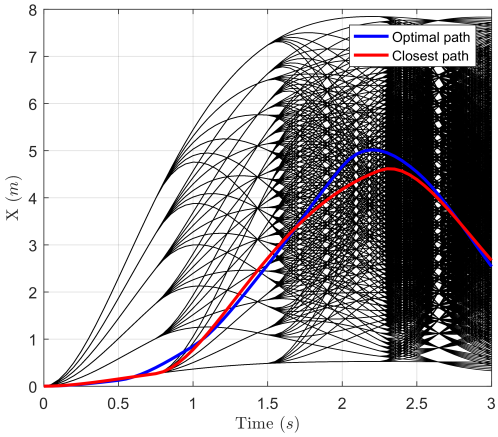
(b) 3 segment, 3 levels



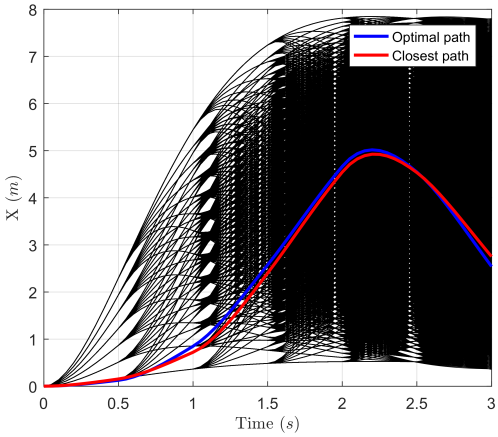
(c) 3 segment, 5 levels



(d) 4 segment, 5 levels



(e) 4 segment, 6 levels



(f) 6 segment, 7 levels

Figure 5.10: Trajectories from sequential input sweeps

against the number of segments and the number of control input quantization levels. It can be clearly seen that increasing both yields ever increasing approximation of the optimal trajectory. This result is not an indication of the "optimality" of the obtained paths, but rather points to the fact that the trajectories generated using sequential control input sweeps approach the case where every possible path is visited.

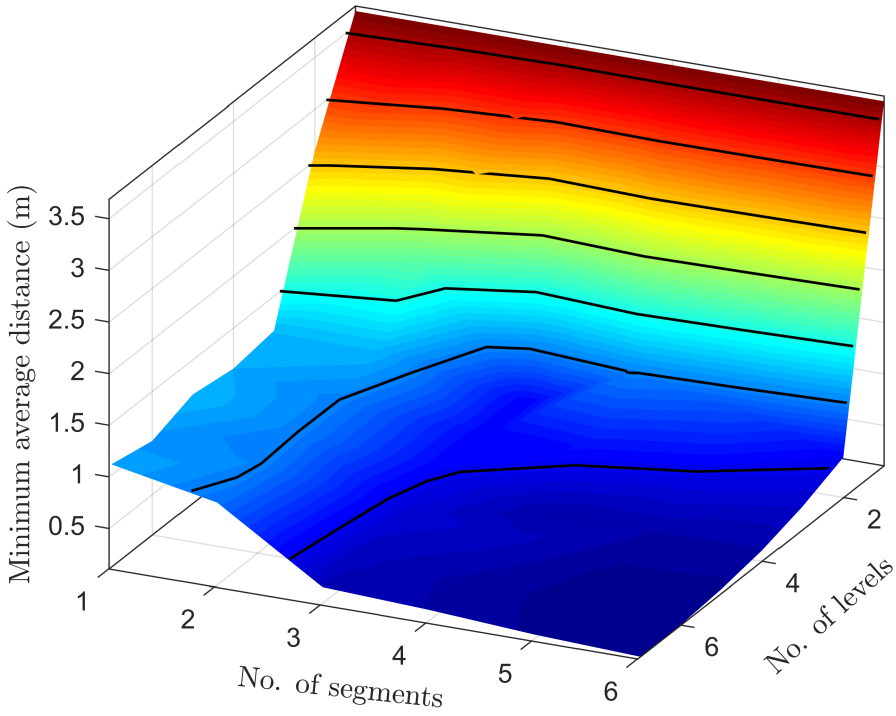


Figure 5.11: Average minimum distance to optimal trajectory

The sequential sweep approach suffers from the curse of dimensionality, which may limit its application for real-world investigations. Even in the 1-D case investigated here, the number of trajectory realizations increases as the number of quantization levels to the power of number of segments - which can grow very quickly in any practical application. In higher dimensions, each additional control channel would raise the number of quantization levels to the power of the number of control channels, so that this number would be exceedingly large in a case with several control input channels. However, the results from this investigation reveal that it is possible to approach an optimal solution, while also

traversing many sub-optimal paths including those resulting from realizations with fixed control inputs across segments. Assuming the computational budget exists to perform a full sequential sweep simulation, any optimal solution should be captured within the generated trajectories. However, if optimality is not a concern, then the fixed control input trajectories still arrive at the limit in the shortest time, which make the simple input sweep an acceptable approach to estimate the minimum time to reach a limit.

5.2.3 Randomized inputs

The sequential sweep is an interesting approach that explores every possible evolution of the system, assuming a finite number of control input quantization levels, and a number of sequential simulation periods. Another approach is to use randomized inputs which can take any value within the admissible range during a particular segment. The Markov property allows the time domain to be split into segments where the control values applied to each trajectory are drawn from a distribution of possible control values. In this manner, the trajectory realizations are randomized, and in the limit would again contain every possible path that can originate at the given initial condition (X and $\dot{X}$ are both zero in the example).

In the simplest case shown in Figure 5.12, a single segment is used with uniformly sampled control input levels. This case differs from the original discrete sweep DOE-style approach only in the fact that the control inputs are sampled from a continuous range rather than at pre-determined quantization levels. Compared to the optimal case, the functional form of the single-segment response is a poor approximation of the response generated with a continuously variable control input, but the response still contains paths which exceed the 5m threshold of the limit parameter. Because the functional form of the constant-input response cannot capture the variability in the optimal path, increasing the number of quantization levels has little effect on the distance between the closest path and the target (Figure 5.13), even with 10000 trajectories.

The next case contains 3 segments where the control input is held constant for the

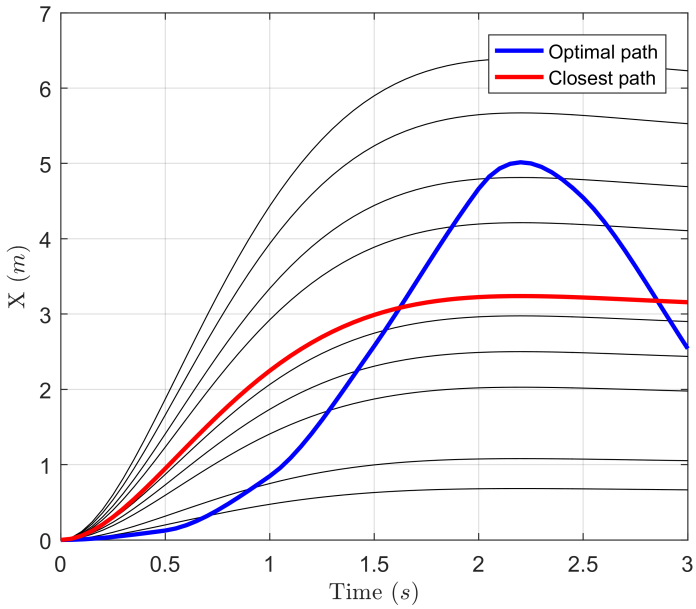


Figure 5.12: Randomized sweep (10 trajectories, 1 segment)

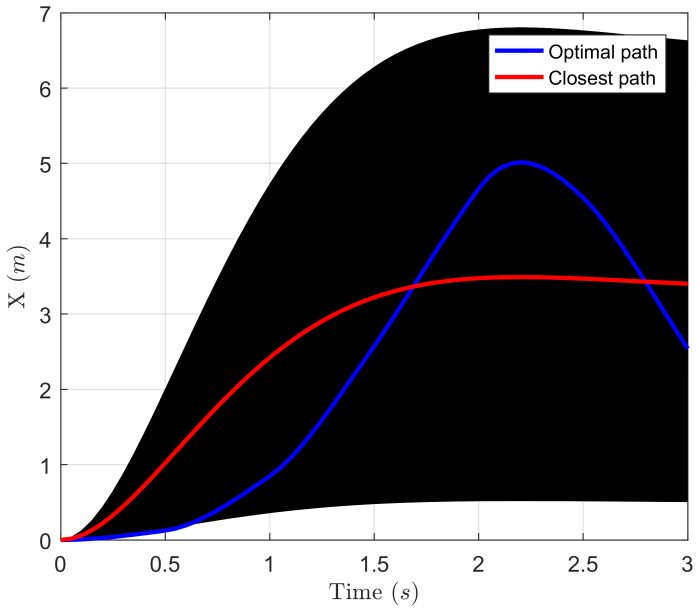


Figure 5.13: Randomized sweep (10000 trajectories, 1 segment)

duration of the segment. At each junction,the inputs are resampled and applied to the ending state of the previous segment. In this manner, randomized realization are generated which display a much greater variability than the constant input sweep. It is interesting to note that the optimal path has resulted in a functional form that is approximated very well when three segments are used. In the case of only 100 trajectory realization, the optimal and randomized paths are quite close, as can be seen in Figure 5.14. Increasing the number of trajectories and further segmenting the time domain results in even finer approximation of the optimal path, though this goal has not been explicitly defined. It is purely a consequence of the random exploration of the trajectories generated using this approach.

These insights are confirmed by inspecting the variation of the distance between the optimal path and the closest realization using the randomized control inputs. Figure 5.16 shows the evolution of the approximation as the number of trajectories is varied between 10 and 10000 and the number of segments between 1 and 12. Two important differences are present in this result compared to the equivalent result in the previous section. The first that in general, increases in the number of trajectories has a similar effect as increasing the number of quantization levels in the previous approach. The second is that while higher numbers of trajectories and segments result in closer approximation of the optimal path, the trend here is less clear.

The surface has been smoothed for visualization purposes and indicates a potential optimum near 6 segments, whereas further increases in the number of segments actually increases the distance between the optimum path and all of the randomized paths. There are several effects that could potentially be causing this result. One is that the functional form of the optimum path happens to be approximated well with a certain number of segments. Another cause is the fact that with finer segmentation, the variability of the generated paths is reduced. This is related to known results for random walk processes, where the variance is reduced over longer sampling periods, compared to the maximum and minimum values that could be achieved given a constant input. Though the random walk result is derived for

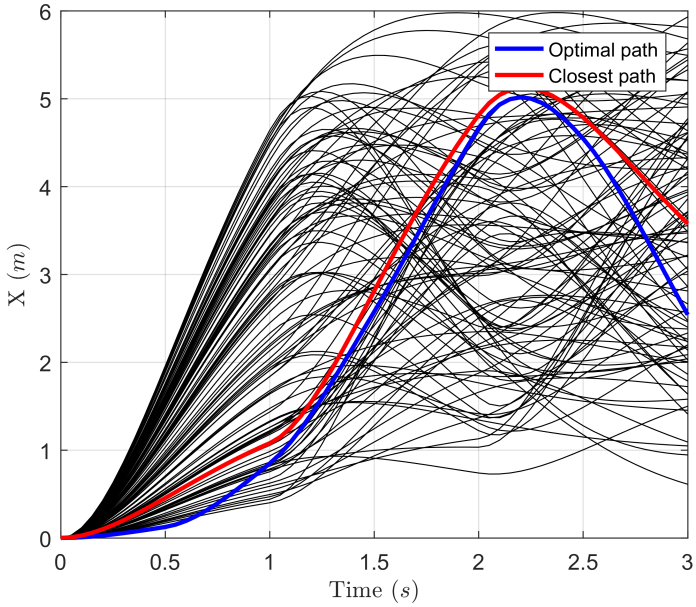


Figure 5.14: Randomized sweep (100 trajectories, 3 segment)

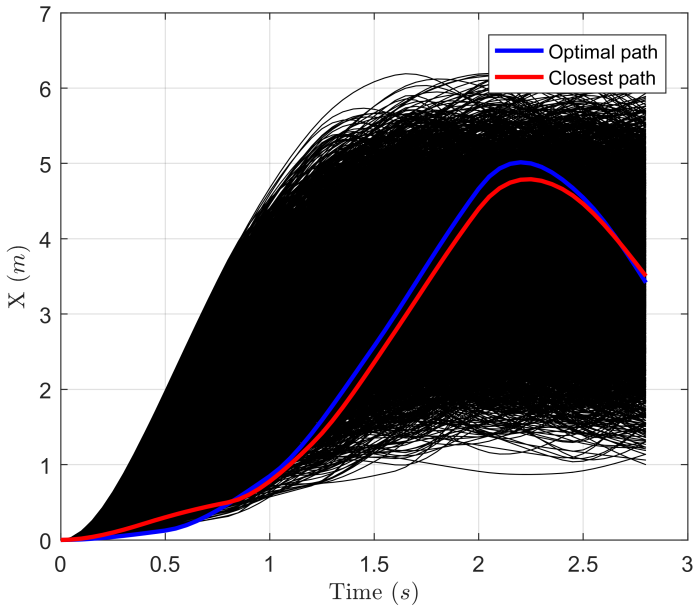


Figure 5.15: Randomized sweep (4000 trajectories, 7 segment)

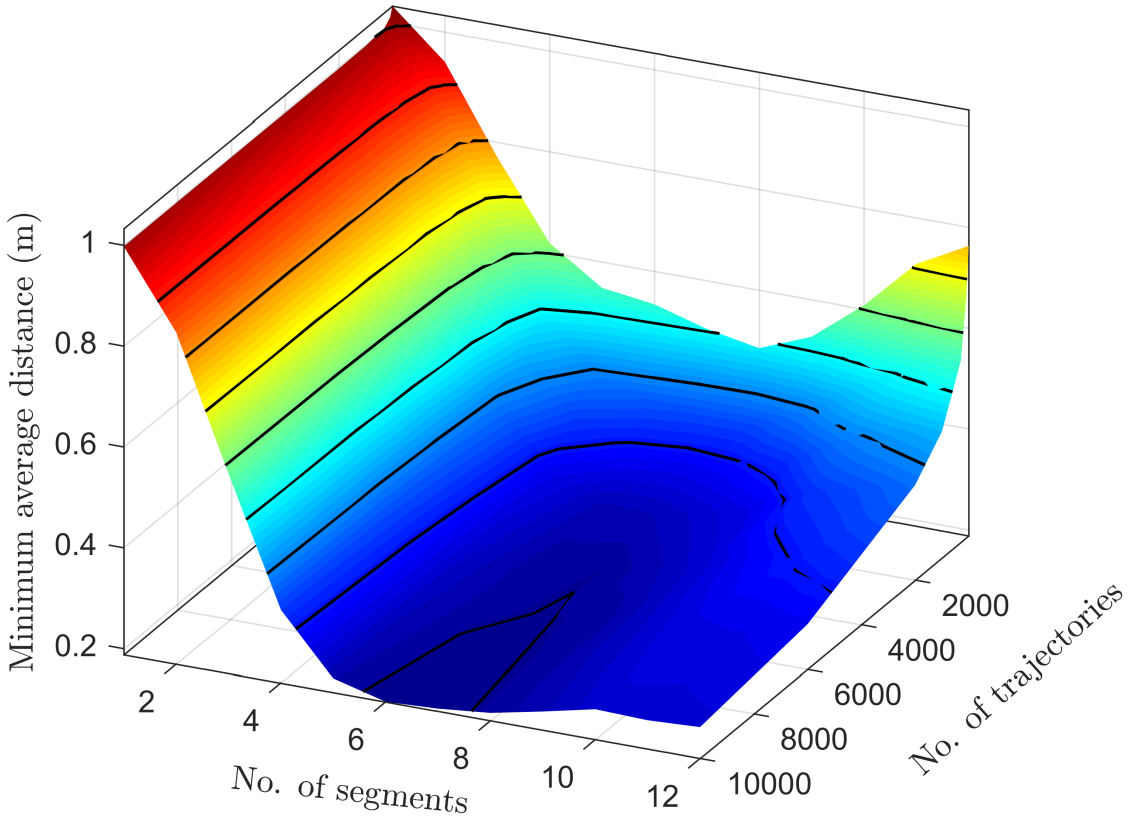


Figure 5.16: Randomized input average minimum distance from optimal

a case where the process jumps to one of two discrete levels, 1 or -1, it is analogous to the manner in which the sampling of the control inputs occurs in the present approach. A final potential cause for the optimum may be related to the fact that the number of segments per unit time is related to the frequency of the input, which would produce the largest responses closer to the natural frequency of the controlled system.

5.2.4 Summary

This analysis showed two additional approaches which warrant further investigation. The sequential sweep is limited by computational considerations and the curse of dimensionality, but it was demonstrated to explore all possible trajectories within the constraints of the number of segments and control input quantization levels. It is interesting to note that given the limit parameter at 5m, the trajectories that reach that threshold the quickest were those where the control input was held constant. The second approach is an analogue to the sequential sweep without the scalability concerns of the sequential sweep arising from splitting each trajectory. Given a computational budget in terms of numbers of samples, the number of segments can be increased in a manner that is appropriate for the problem at hand, keeping in mind the insights regarding the optimum number of segments. The desired frequency of the input, the variance reduction with higher number of segments, and the functional form of any desired responses should be considered.

Therefore it can be concluded that both alternatives are appropriate from a purely "exploratory" perspective. However, issues arise if the outcome of the simulations are to be mapped to the input domain. Specifically, in both the random control approach and the sequential sweep, mapping the initial condition to the final state after several segments is not straghtforward, since the controls trajectories overlap and intersect during the simulation. The mapping requirement necessitates a clear relationship between inputs and outputs, which is possible using a discrete sweep.

The single input sweep approach cannot generally approximate complicated trajecto-

ries, but is frequently associated with the worst case scenario. Given the ability to directly map the input values used in the simulation to the risk of the associated response is an important benefit if the goal is to establish effective detection thresholds in a monitoring environment.

Additionally, the discrete sweep is identical to the first segment of the sequential sweep approach. In this regard it could mean that the simulation, once started at a pre-defined initial condition, can propagate throughout the range of potential outcomes and identify "all" possible scenarios. However, the rapid rise in the number of simulated trajectories means that the sequential sweep approach is impractical for longer time horizon predictions if applied without some means to control the increasing variability. A way to mitigate this problem, aside from extreme amounts of resources, is to limit the number of redundant intermediate states that are visited, which in the original sequential sweep approach are not being tracked. By starting a discrete sweep at a new condition close to the initial start state, an additional set of solutions can be created in the vicinity of the condition of interest. After repeated application of the discrete sweep, an ever closer approximation of the sequential sweep can be achieved, while avoiding the computational explosion of the original sequential sweep approach. Therefore the discrete sweep is attractive for use in the dynamic simulations.

The randomized input alternative was shown to also approximate every possible trajectory, but with some additional and important caveats. In the sequential sweep case, increasing the number of segments per unit time results in more variability, whereas the variability is not monotonically increasing in the same manner for the random inputs case, as was shown in Figure 5.16. In both the sequential sweep and randomized input cases, however, increasing the number of trajectories results in increasing exploration of the operational space. One major drawback of the randomized input approach is that the relationship between the inputs and outputs is no longer preserved, so a mapping cannot be constructed directly between the inputs to the simulation and the outcomes. Therefore the answer to

the hypothesis to RQ4 is that while the randomized input approach is great for exploration, the repeated application of the discrete sweep at an increasing range of initial conditions will approximate the solution of the sequential sweep while preserving the mapping ability - yielding a set of trajectories which can be represented using a supervised learning technique and implemented as a monitoring boundary.

5.3 2-D and 3-D Tipover analysis

The purpose of this experiment was to determine the feasibility of the overall approach of using physics based models for the definition and eventual monitoring of hazardous flight conditions. The main goal is to address the question of whether the models can be effectively used to describe boundaries, which can be related to a critical parameter limit which can be objectively defined. The tipover scenario is hazardous because of the potential for contact with the ground and associated damage. The first part of this section deals with a condition specific model that was used to establish the basic approach and build experience with the method's utility in defining events based on the simulated response of the vehicle. This model helps answer the question regarding the general framework of using offline simulations to identify parameter limits for monitoring, and provides some insight into the type of response that can be achieved so that the appropriate choices for output representation can be made.

5.3.1 2-D Dynamic Model for Tipover analysis

This section documents a 2-D dynamic model used in an initial demonstration of the basic concepts within the present methodology. The idea is that if the dynamics can be modeled faithfully, a simulation approach could be devised that will investigate the accident conditions ahead of time and provide guidance to inform the development of any protective measures. Specifically, the G-TIGT accident resulted in the implementation of limits on the combinations of control inputs that correspond with increased risk of rollover. If the

simulation can reveal if such control inputs exist, and where they occur within the range of inputs possible for the given rotorcraft, then the potential exits that the knowledge about the proximity of the hazard would allow an operator to take corrective measures.

A simplified lateral dynamics model similar to those used by Fox[105] and Blackwell[129] is used for the purpose of studying the lateral behavior of a helicopter on the ground. The model contains a rigid body acted upon by forces through the landing gear struts as they come in contact with the ground, and two force vectors representing the main rotor and tail rotor forces. Additionally, a the main rotor force vector can be tilted, which generates additional moments due to a torsional spring at the "hub". An overview of the model and its components is shown in Figure 5.17. The two main landing gears legs are modeled as oleo struts with flexible ground contact points to simulate tire deflections in the ground plane. The following development is presented for one of the main gears. The force in the oleo strut is calculated using Equation 5.8, following a similar development in [129].

$$L_i = \begin{cases} K_1 {d_i}^2 + C_1 \dot{d_i} & (d_i < d_o) \\ K_1 {d_0}^2 + K_2 (d_i - d_0)^2 + C_2 \dot{d_i} & (d_i \geq d_0) \end{cases} \tag{5.8}$$

In Equation 5.8, L_i is the force due to compression of the oleo, d is the oleo deflection, and d_0 is the oleo break-point as defined in [133]. The subscript i indicates which landing gear strut the calculation pertains to. In this 2-D model, there are only 2 landing gear struts, marked as landing gear a for the left one and b for the one on the right, but helicopters with three or more landing gear legs exist.

The force on the each strut is generated when the landing gear is in contact with the ground. The ground contact point is initially determined as the point of intersection between the ground plane and the vector which originates at the strut attachment point on the fuselage and is oriented along the fuselage y direction (in 3D this would be the positive z direction). Following initial contact with the ground, the contact point moves according to the change in fuselage position and orientation. Tires generate forces which causes them

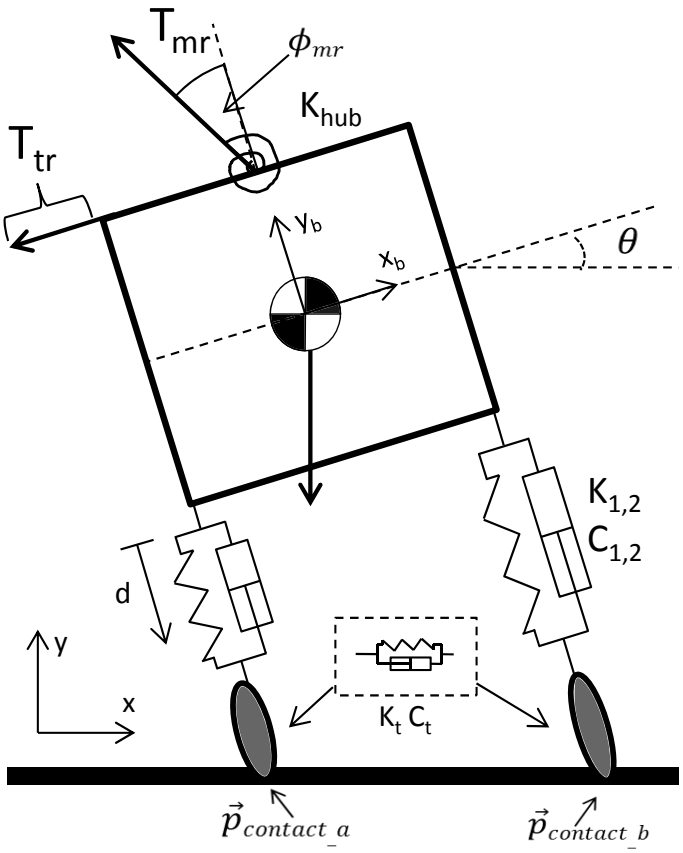


Figure 5.17: Schematic of the 2D tipover model

to deflect, and this deflection is considered using another point which represents the contact patch, the actual point of contact between the "tire" and the ground. The lateral force depends on the component of tire deflection in the ground plane, and the rate of change of this deflection, as shown in Equation 5.9.

$$F_i^{\text{tire}} = -K_t(x_i^{\text{patch}} - x_i^{\text{contact}}) - C_t\dot{x}_i^{\text{contact}} \tag{5.9}$$

The location of the contact patch is propagated as a model state when the equations of motions are integrated in time. The contact patch location does not change when the friction force is below the static friction limit, and follows the lateral motion of the strut contact point when the static friction limit is exceeded. The force transmitted to the strut due to the

tire is thus found as in Equation 5.10.

$$Fx_i = \begin{cases} F_i^{\text{tire}} & |F_i^{\text{tire}}| < \mu Fy_i \\ \text{sign}(F_i^{\text{tire}})\mu Fy_i & |F_i^{\text{tire}}| \geq \mu Fy_i \end{cases} \tag{5.10}$$

The vertical force at the contact patch Fy_i is first calculated for the no-slip condition, and is modified if slip is detected.

$$Fy_i = \cos\theta L_i \tag{5.11}$$

$$P_i = \cos\theta Fx_i \tag{5.12}$$

The Equations 5.11 and 5.12 yield the forces experienced by the end of the landing gear strut over the range of interest. The current analysis assumes a critical angle (60deg.) at which point the simulation is stopped. Summing the contributions and resolving them in the inertial frame results in the following contribution of the first landing gear strut to the total force acting on the fuselage of the vehicle, where R is the rotation matrix that rotates the body frame to the earth x-y frame by angle θ:

$$\boldsymbol{F}_i^{\text{LG}} = R \begin{bmatrix} P_i \\ L_i \end{bmatrix} \tag{5.13}$$

The moment contribution of the i_{th} gear is calculated using 5.14:

$$M_i = (\overrightarrow{\boldsymbol{r}}_{\text{contact}} - \overrightarrow{\boldsymbol{r}}_{\text{cg}}) \times \boldsymbol{F_i}^{\text{LG}} \tag{5.14}$$

The main rotor and tail rotor are represented as force vectors with a first-order response to magnitude commands. Additionally, the main rotor force vector can be tilted by an angle ϕ in response to cyclic inputs, which causes a deflection of the rotational spring at the hub.

The main rotor tilt is also modeled as a first-order response.

$$\boldsymbol{F}_{\text{rotors}} = R \begin{bmatrix} T_{TR} - \sin(\phi) T_{MR} \\ \cos(\phi) T_{MR} \end{bmatrix} \tag{5.15}$$

$$M_{\text{rotors}} = (\overrightarrow{\boldsymbol{r}}_{\text{hub}} - \overrightarrow{\boldsymbol{r}}_{\text{cg}}) \times \boldsymbol{F}_{\text{rotors}} + K_{\text{hub}}\phi \tag{5.16}$$

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} = \frac{1}{m} \left(\overrightarrow{\boldsymbol{g}} + \boldsymbol{F}_i{}^{\text{LG}} + \boldsymbol{F}_{\text{rotors}} \right) \tag{5.17}$$

$$\ddot{\theta} = \frac{1}{I} (\sum M_i + M_{\text{rotors}}) \tag{5.18}$$

Table 5.1: 2-D tipover model parameters

Model parameter	**Value**		**Description**
W	3	m	fuselage width
H	4	m	fuselage height
m	3000	Kg	mass
$d0$	1	m	undeflected strut length
I	6.25 103	Kg m2	moment of inertia
K_1	1.0	N/m	Oleo spring const.
K_2	1.4	N/m	Oleo spring const.
C_1	1.5	N s/m	Oleo damping const.
C_2	1.8	N s/m	Oleo damping const.
K_t	2.2	N/m	Lateral tire stiffness
C_t	3.0	N s/m	Tire damping const.
τ_{MR}	0.7	s	Main rotor force time const.
τ_{TR}	0.3	s	Tail rotor force time const.
τ_ϕ	1.3	s	Lateral “flapping” time const.
K_{hub}	1.95 104	N m/rad	Hub spring stiffness

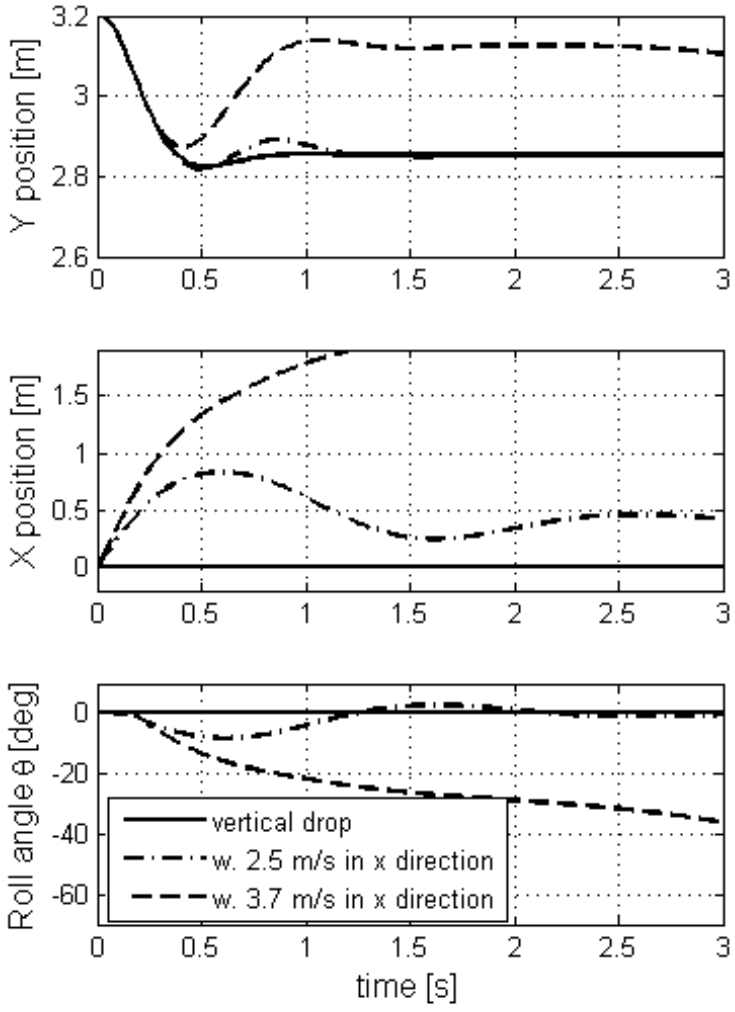


Figure 5.18: Result of 2D tipover simulation

5.3.2 Experimental setup

The test was set up in a the same manner as described later in section 5.5 describing the FLIGHTLAB implementation of the approach. The lateral cyclic and tail rotor pedal were varied for a range of input values, with the result being a trajectory of the vehicle states for a specified length of time. Each of these trajectories is associated with a unique combination of control inputs. The result is then cast in the form of the first hitting time of the limit parameter. Note that there were no simulated contacts between the vehicle and the ground in either of these cases, but rather a critical angle is assumed and the simulation stopped when this angle is reached. For the 2-D analysis, an angle of 60 degrees was specified. This is far beyond what would be experienced in practice. For the 3-D model, a lower number was chosen to take into account the realization that the main rotor would come in contact with the ground at moderate angles of roll. Figure 5.19 shows the manner in which the roll angle was tracked and used to determine the first hitting time in both cases.

5.3.3 Analysis

Figure 5.20 shows the output of one such analysis using the 2-D model the main rotor thrust was set at a moderate setting of about two thirds the vehicle weight, representing a ground maneuvering case. The lateral cyclic controls the tilt of the main rotor thrust vector, which in combination with the tail rotor, produces momenta that cause the vehicle to roll over. The time to reach the critical condition is shown in this figure as a progression from a safe region in the center of the figure, representing trim values, and achieving the critical condition in decreasing lengths of time as the control input increases. This initial result is promising from the standpoint that the analysis was performed without assuming a form of the boundary determined by the simulation runs, and the output contains the entire spectrum from perfect safety to very little time to a crash.

Comparisons with the original detection threshold following the G-TIGT accident shows that there is a discrepancy between this result that stems from the fact that helicopters are

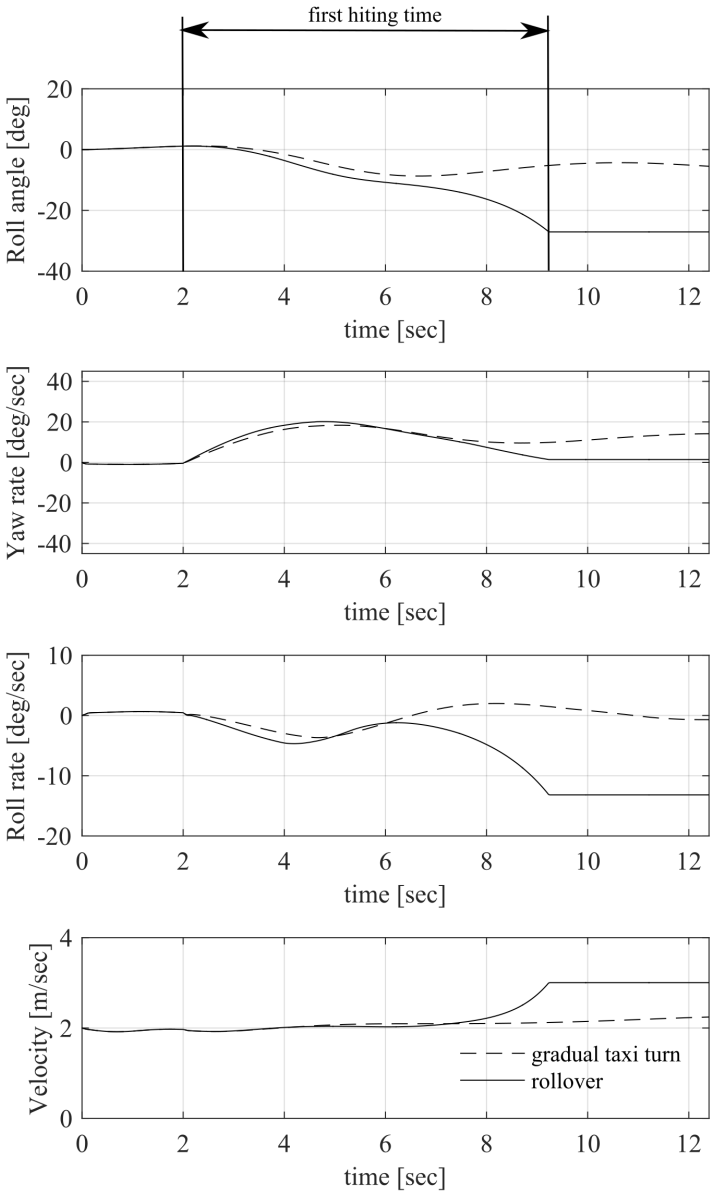


Figure 5.19: Time to critical roll angle

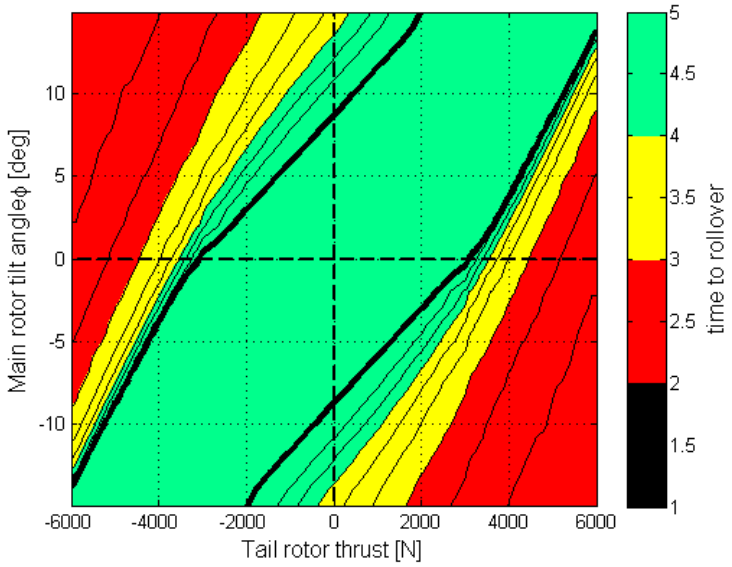


Figure 5.20: 2-D model result

asymmetric vehicles, whereas the 2-D test case is a simplified, symmetric conceptual representation of the helicopter on the ground. In addition, many other effects are not considered in this simple case. The procedure was therefore repeated using a 3D dynamic model. The components of this model were described in the section on nonlinear helicopter modeling. The components of this model are well established in literature, with the primary difference being how the ground contact is modeled given the ability of the model to roll on the ground, as opposed to the simple 2-D model. The mapping shown in Figure 5.21 differs from the 2-D result in several important areas. One is that the response map is no longer symmetric about the diagonal, due to the asymmetry of the vehicle. While the absolute values of the first hitting time seem reasonably close to those obtained with the 2-D model, there are regions in the corners of the domain which show an unexpected behavior. Though the inputs are at the extremes of the domain, the time to reach the critical condition is longer compared to some more moderate inputs. Two primary observations stem from the experience of modeling tipover with dynamic models. The first is that boundaries can be objectively defined ahead of time using a structured approach. The second is that to capture the full complexity of the helicopter response, a 2-D model may not be sufficient and

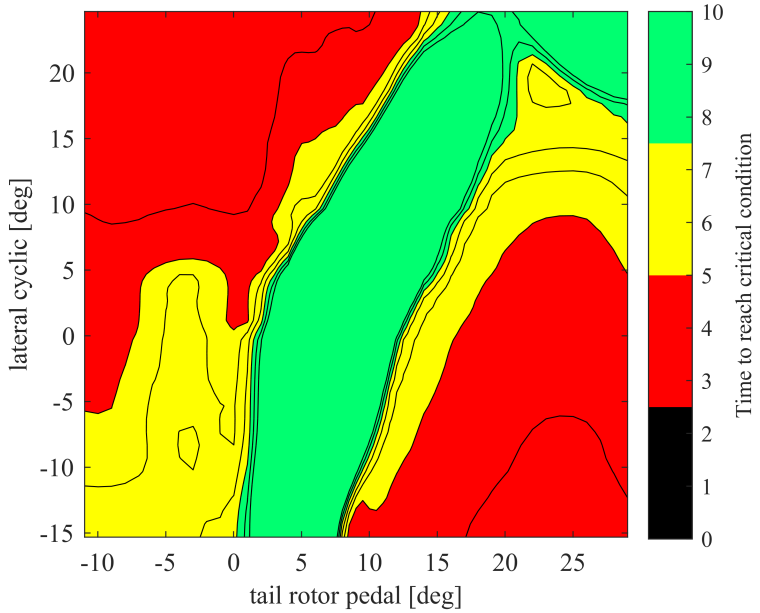


Figure 5.21: 3-D model result

a detailed 3-D dynamic simulation is required. The 3-D model used for this test case is of this type, but the final test case of this work and any real-world application would certainly use a validated and industry standard simulation such as FLIGHTLAB.

This experiment is a continuation of the work in support of the Hypothesis to RQ3, with an emphasis on detecting critical conditions prior to their explicit definition by an expert. Additionally, the results differ between the two model fidelity levels, pointing to the need to incorporate as many of the known physical phenomena and effects as is practical for any given problem.

5.3.4 Implementation of dynamic monitoring boundary

The mappings shown in Figures 5.20 and 5.21 can be used directly as guidance to the operator, but the main benefit is obtained by establishing monitoring limits based on the time to reach a critical state. In the case of the 2-D analysis, this critical state is a rollover of the vehicle (taken to be 60 degrees roll angle). This subsection discusses the manner in which monitoring limits are established with regard to the operator's preference and

tolerance for risk, given the output of the dynamic simulation.

After the output is generated and the mapping between input conditions and the time-based criterion is established, the results are represented using a neural network parametrization (discussed in section 5.4). This multi-dimensional surface can then be implemented in a database in order to calculate the time to critical condition given an input vector from the flight data. Figure 5.22 shows a subset of this parametrization at the nominal roll angle and medium main rotor thrust (20kN) setting for the 2-D example, demonstrating accurate representation of the actual surface (on the right) using the neural network (left surface). This neural network representation of the input-output mapping can then be directly im-

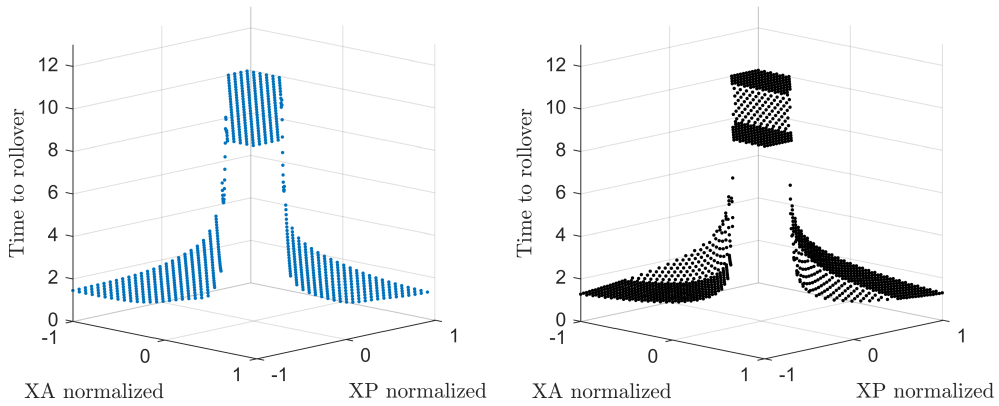


Figure 5.22: NN fit for 2-D example

plemented in an HFDM database, providing an indication of the time to a critical state given the input data. Individual operators might have a different tolerance for risk and would prefer to be alerted with different times to the critical condition. The NN representation of the input-output map can be used to establish monitoring limits, which when exceeded by the relevant parameters would trigger an alert in the HFDM system. Figure 5.23 shows an example of how this type of limit would be established given two different levels of the time based criterion. The neural network fit is sliced using level surfaces at the operator-determined time. In this figure, two horizontal surfaces are shown at 8s and 3s on the vertical axis, representing the alerting thresholds that might be set by two different

operators. The resulting level curves (shown as thick black lines on the NN surface) would be the typical monitoring threshold on the selected parameters that operators would use in practice.

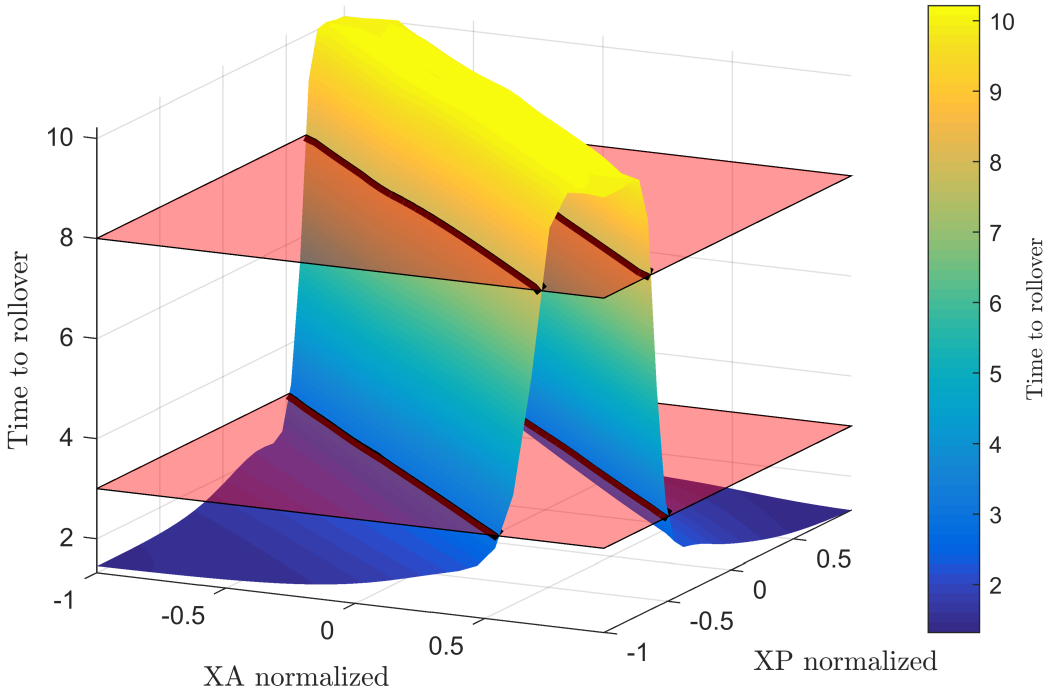


Figure 5.23: Establishing monitoring thresholds using a time criterion

The monitoring threshold is thus only used for alerting the operator, while the entire mapping is available to the HFDM system to determine the extent of the breach. Furthermore, the monitoring thresholds shown by the level curves would normally be implemented as static boundaries, defined and refined over time, but independent of the instantaneous flight condition. Using the present approach a range of input conditions can be explored and the entire set of resulting times to a critical limit to be represented using the neural network.

As an example, the neural network representation of the time to rollover is developed for a range of roll angles and thrust settings around the nominal condition. When an operator sets an alerting level (in terms of time), the resulting boundaries resulting from the

intersection of the mapping with the level curve at that alerting level account for this variation, and allow the boundary to change dynamically in response to control inputs or the operating condition (roll angle in this example). Figure 5.24 shows curves set at 3 seconds to rollover, as they shift in response to the present roll angle. Thus as the vehicle roll angle increases, the indication of time to the critical boundary changes.

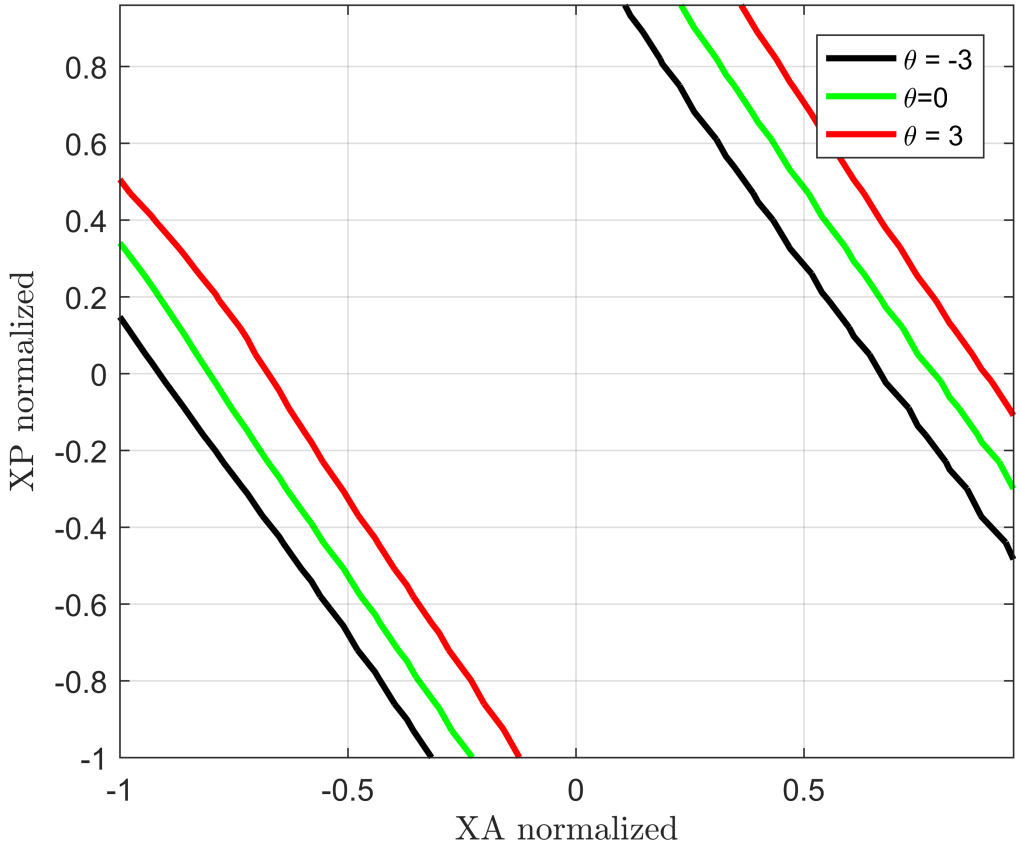


Figure 5.24: Variation of alerting threshold with roll angle

Figure 5.25 shows the variation of the 3 second thresholds with different main rotor thrust levels. As a result of increased main rotor thrust, the region within the 3 second boundary is reduced, meaning that smaller control inputs will result in a rollover as the thrust is increased. This figure only shows 3 settings for clarity, but the entire range between these values is represented using the neural network, as well as at all other roll angles and control values represented in the simulated dataset.

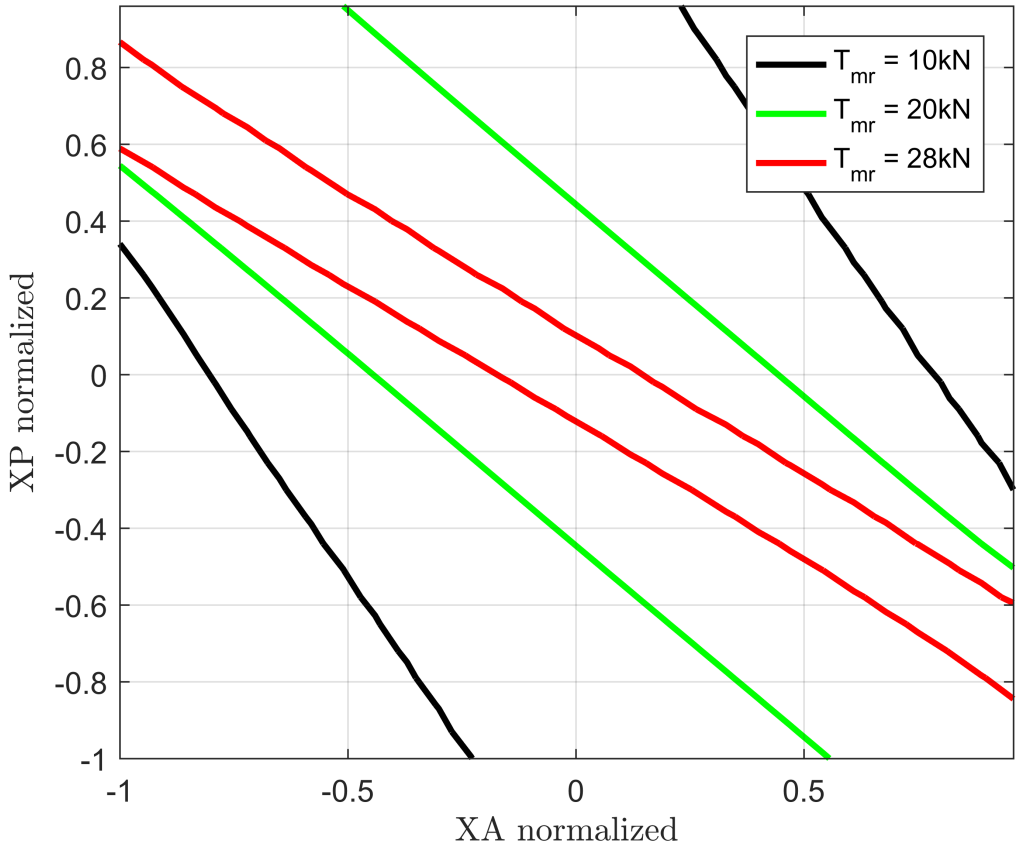


Figure 5.25: Variation of alerting threshold with roll angle

5.4 Output representation

The output resulting from the dynamic simulation is a set of time histories for each modeled parameter, which are then post-processed to determine the first hitting time for any parameter of interest. If the inputs are known, and the result associated with each input can be determined, then a mathematical representation of these relationships can be defined. In the present investigations, the insight gained from the experimentation in section 5.3) helped illustrate the type of result that could be expected from a simulation run for a specific operating condition. Given the response of many parameter and their associated time histories, the mappings constructed in section 5.3 are the result of plotting any limit parameter against the input vector and monitoring the time to reach any one of them, resulting in highly nonlinear surfaces. The nonlinearity partly arises out of the fact that the helicopter can leave the ground and come in abrupt contact with it, in addition to all other nonlinear aspects of the helicopter response.

Based on a review of existing methods, the Multi-Layer Perceptron was chosen as the baseline neural network architecture to be used for the purpose of representing the simulation result in a monitoring setting. This choice is motivated by the general ability of neural networks to model complicated nonlinear aspects of the data, and their demonstrated use in large datasets, some involving aviation data. This section evaluates neural networks as a potential solution to represent the results of the dynamic simulation studies in a consistent manner that lends itself to monitoring applications. If the simulation results can be adequately represented by a neural network surrogate, monitoring of the hazards determined in simulation would be enabled. Additionally, the operator would retain the ability to set a critical time for monitoring purposes, and the requirements to store data would be reduced, further increasing the potential benefits from the application of the DEMMoS method.

5.4.1 Experimental setup

In Figure 5.26, four test surfaces are shown which were used in this evaluation. The surfaces were generated to represent the type of characteristics expected in the output data, while limiting the number of input dimensions to 2 for visualization purposes. Each surface can be described as follows:

1. Smooth ridge with a linearly varying mean

$$y(x_1, x_2) = \frac{10}{\sqrt{2\sigma^2\pi}} \exp\left(-(x_1 - .1x_2)^2/(2\sigma^2)\right) \tag{5.19}$$

   where $\sigma = 3$, and the independent variables x_1,x_2 were in the range $[-15, 15]$ which were normalized to $[-2, 2]$ for the purpose of performing the neural network regression.

2. Smooth ridge with a sinusoidally varying mean as a case of a highly nonlinear surfaces with characteristics that vary significantly along each input variable.

$$y(x_1, x_2) = \frac{10}{\sqrt{2\sigma^2\pi}} \exp\left(-(x_1 - \sin(x_2))^2/(2\sigma^2)\right) \tag{5.20}$$

3. Smooth Gaussian peaks for situations when the response is multi-modal.

$$y(x_1, x_2) = \sum \frac{\exp\left\{-\frac{1}{2}\left[\left(\frac{x_1-\mu_{x_1}}{\sigma_{x_1}}\right)^2 + \left(\frac{x_2-\mu_{x_2}}{\sigma_{x_2}}\right)^2\right]\right\}}{2\pi\sigma_{x_1}\sigma_{x_2}} \tag{5.21}$$

   where no correlation between the variables was assumed and the standard deviations were equal at $\sigma_{x_1} = \sigma_{x_2} = 3$. The means for the two peaks were at $[5, 6]$ and $[-10, -5]$ before normalization.

4. Truncated ridge with a sinusoidal mean variation.

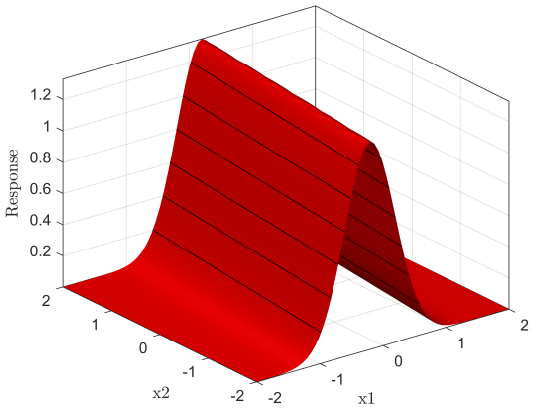
$$y(x_1, x_2) = \frac{\exp\left\{-\frac{1}{2}\left[\left(\frac{x_1-\mu_{x_1}}{\sigma_{x_1}}\right)^2 + \left(\frac{x_2-\mu_{x_2}}{\sigma_{x_2}}\right)^2\right]\right\}}{2\pi\sigma_{x_1}\sigma_{x_2}} \tag{5.22}$$

where $\sigma_{x_1} = 3$, $\sigma_{x_2} = 30$, and $\mu_{x_1} = \sin(x_2/2)$, $\mu_{x_2} = 1$. Note that the maximum response $y(x_1, x_2)$ was normalized to a maximum of 1.5 before being truncated at 1. The result is a complex surface that contains a sharp edges, a region at the top with a perfectly flat surface, and a sinusoidal variation. This is typical of the type of surfaces obtained from post-processing the simulation results.

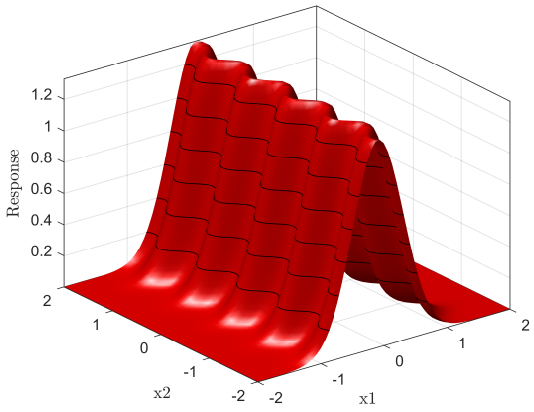
These expressions were used to generate the training data, so that the different neural network options investigated can be compared. The data were normalized to the same $[-2, 2]$ range for both input variables in all four cases. The data created using these surfaces were used to train different neural network architectures to evaluate their applicability in modeling the simulation outputs and gain a perspective on the characteristics of the models.

The next step is to select a neural network architecture, number of neurons, and to train on the data. The network architecture involves several choices, including the activation function used for the neurons and the number of layout of layers and connections. For this investigation, a Multi-Layer Perceptron (MLP) was selected due to the nonlinear and complex nature of the outputs. Of the possible network topologies, a commonly used model is the single hidden layer (SHL) network, with an input and an output layer connected to the hidden layer. The layers are fully connected, meaning the nodes in each layer connect to each node of the adjacent layer, but without any connections within a layer. In addition to the SHL, multiple hidden layer architectures were also considered due to the increased flexibility in representing complex surfaces, especially given the presence of edges and other detailed features in the data.

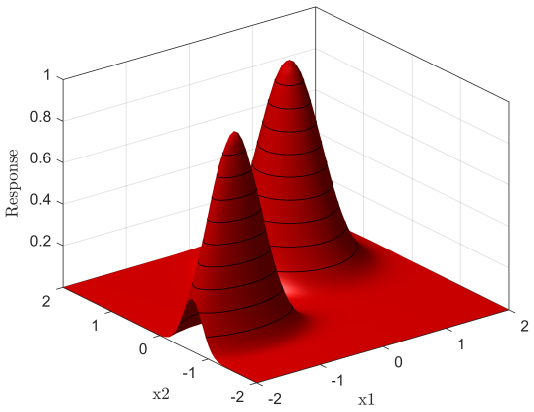
The activation function is another design choice when creating a neural network. Based on the problem, a different neural network may be appropriate. For example, in large



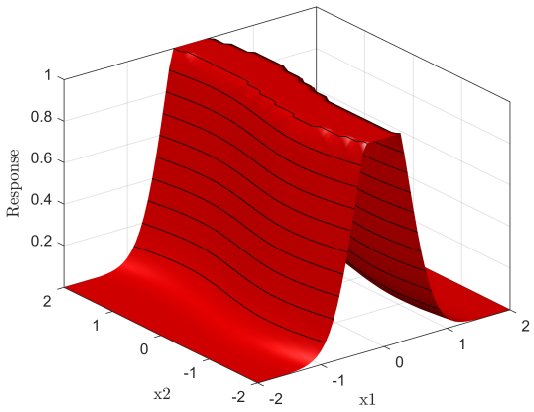
(a) Smooth ridge with skew



(b) Smooth ridge with sinusoidal mean



(c) Gaussian peaks



(d) Truncated ridge with sinusoidal mean

Figure 5.26: Test surfaces

networks with many hidden layers and nodes, the rectifier linear function is frequently used because it offers a great deal of flexibility in representing structures in the data. This has become especially useful in image analyses and deep learning tasks. For the problem in this work, the datasets are much smaller and it is desirable to keep the number of neurons to a low value, though there is no specific requirement. With fewer neurons, it is easier to obtain smooth approximations using one of the other activation functions, such as the sigmoid or $tanh$ function. The sigmoid and $tanh$ have a smooth derivative, which is used by training algorithms such as gradient descent during backpropagation, a typical network training method. However, the complex interplay between the characteristics of the dataset, the network architecture and the target variables means that different applications may benefit from a particular algorithm. This is especially important when there are numerous hidden layers, as this usually leads to vanishing gradients which can stall the improvement during network training. In this work, a quasi-Newton algorithm was used, which approximates the gradient at each layer based on past values. The algorithm handles multiple layers well in cases with a moderate number of input variables, which is certainly the case here. The performance of a neural network can be measured using different metrics depending on the application. In the present case, the intended use is regression, so standard metrics such as sum of squared errors, mean sum of squared errors, R^2 and qualitative plots are applicable. The mean squared error is defined as:

$$MSE = \frac{\sum_{i=1}^{n}((\hat{y}_i(x_1, x_2) - y_i(x_1, x_2))^2)}{n} \tag{5.23}$$

where $\hat{y}(x_1, x_2)$ is the neural network estimate of the response y, and n is the number of sampled points. The squared sum of errors is also used, which is then dependent on sample size but is useful in the case where the sample sizes are equal, as is the case here. It is also quite telling to examine the actual set of responses against the predicted values, which may indicate properties in the data that could be useful in determining the network

characteristics.

In summary, the experimental procedure for the evaluation of the neural network regression approach is as follows:

- Step 1: Define test surfaces.
- Step 2: Determine range for the number of hidden nodes.
- Step 3: Perform network training using appropriate algorithm for each network size (including [2-8-8-2-1]).
- Step 4: Validate fits using separate set of test data.
- Step 5: Analyze fit metrics and visually assess the regression result.
- Step 6: Select best network architecture.

In this analysis, the SHL was evaluated for goodness of fit in the standard bias-variance trade off. It is reasonable to expect that an optimum number of nodes can be identified. However, it is clear from the results that the MLP with multiple hidden layers is a superior candidate to the SHL for regressing complicated surfaces such as those output from the dynamic simulation and first hitting time estimation.

5.4.2 Analysis of neural network fit characteristics

The first investigation is to compare the ability of a SHL neural network to approximate the surfaces shown in Figure 5.26. The $tanh$ activation function was used in each case. The alternatives were also briefly evaluated but it was quickly determined that $tanh$ was appropriate for the types of surfaces expected as output from the dynamic simulation and that were created as test cases for this experiment. To perform the experiment, the number of hidden nodes was varied and the fit metrics were tracked. Each result is visualized to provide a perspective on the types of surface features that are fit easily by the network, and to potentially indicate aspects which would require refinement. The number of nodes

was varied between 1 and 10, along with an additional case with 100 nodes. The goal is to establish whether the network fit is improving with increasing numbers of nodes, and if not, then to propose changes that will improve the fit.

Test case 1: Smooth ridge

The first case is a smooth Gaussian ridge with a slight skew in order to make it multivariate. If the mean of the ridge is held at a constant $x1$ value, the regression would become one-dimensional, and the goal in this investigation is to evaluate the ability to approximate multi-dimensional surfaces. The number of nodes was varied and excellent fits were obtained using all test networks. Figure 5.27 shows the original surface in red with points generated by the fitted neural network. Only five hidden nodes were used, and the resulting points are almost indistinguishable from the actual surface.

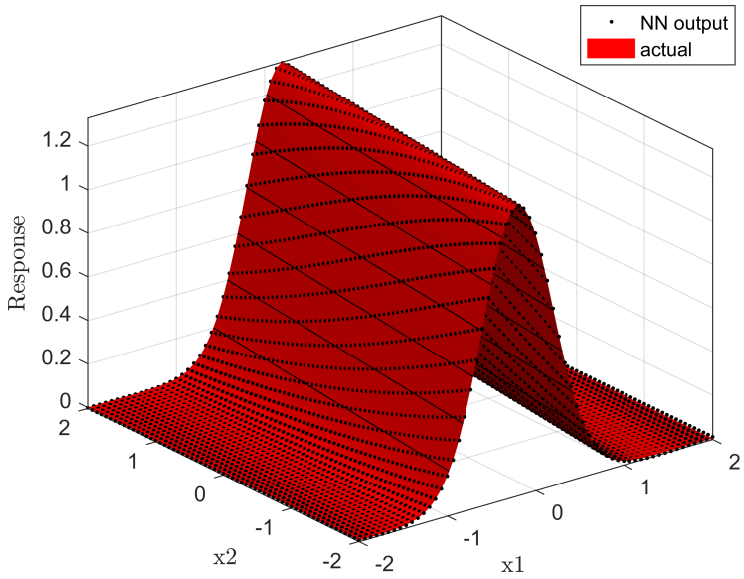


Figure 5.27: Smooth ridge fitted with 5 hidden nodes

In addition to inspection, the prediction can be visualized by looking at the actual-by-predicted plot, which shows a near-perfect fit. Indeed for the smooth ridge constructed using a single Gaussian and a linearly varying mean, the SHL neural network provides

very accurate fits with fewer than 2 nodes or more. The reason for this outcome is that the activation function can easily approximate some of the features of a ridge, especially since this particular case is very close to being one-dimensional. A single $tanh$ node cannot capture the surface, but with two nodes the fit is already close to perfect. This can be visualized by summing the contributions of two $tanh$ functions, one of which has been reversed and shifted - which is approximately what is achieved during the training of the network.

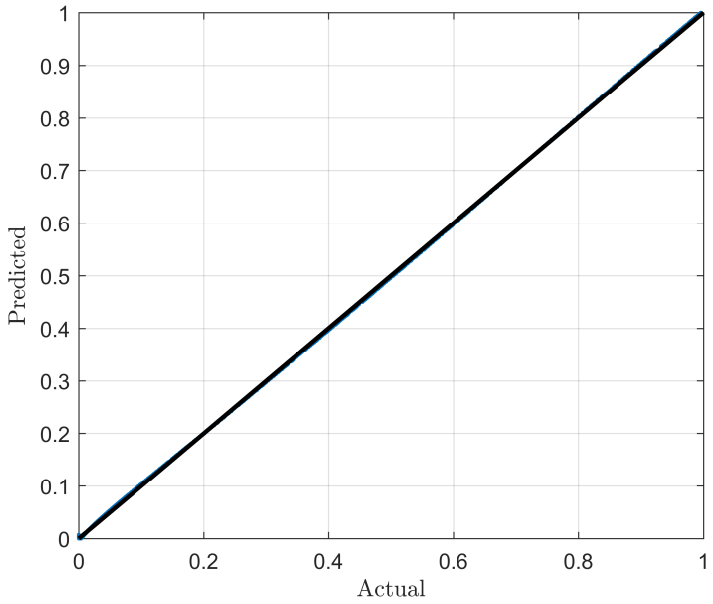


Figure 5.28: Estimated versus actual response with 5 hidden nodes

The training held back 10% of the available data for testing use during the training of the network. However, the full application of the neural networks will use results from dynamic simulation using FLIGHTLAB, where the primary data will not contain redundant data points. Holding back any proportion of points for validation may cause the training to miss important features of the multi-dimensional surface, especially in the presence of edges. For the smooth ridge in this first case, this is not a problem, but it was observed to be an issue in the following cases. A separate dataset validation dataset is therefore generated over the same range of input variables for testing purposes.

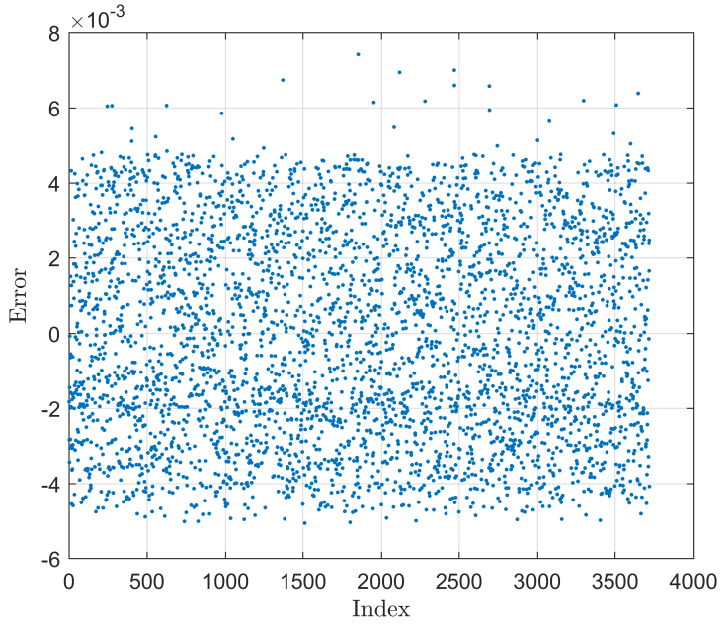


Figure 5.29: Residuals for smooth ridge with 5 hidden nodes

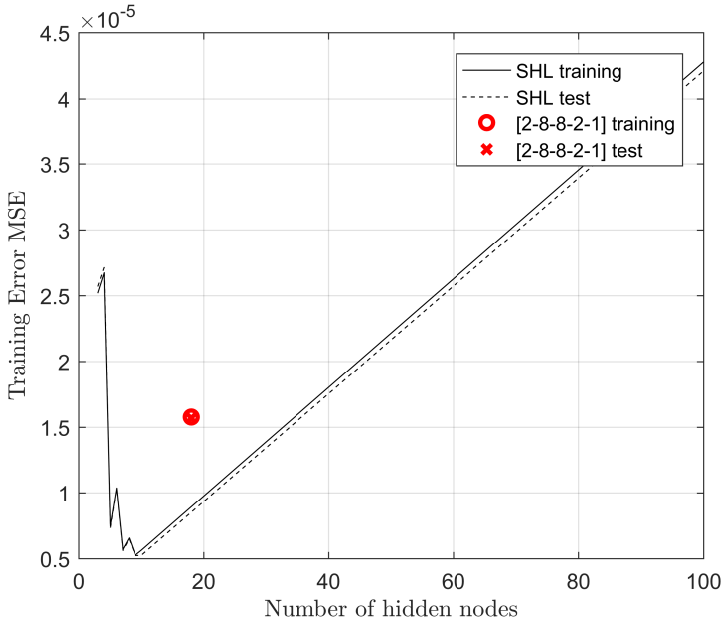


Figure 5.30: MSE with varying network size

Figure 5.30 shows the mean squared error of the neural network fit across several network sizes in terms of numbers of hidden nodes. Errors for both training and test datasets are shown. The graph shows a clear trend toward an optimum at a moderate number of nodes, but the exact optimum is not shown or calculated during this test due to the fact that the fits is adequate for essentially any of the tested network sizes. The last data point is for a network with 100 hidden nodes, which has poorer performance than the smaller networks.

In addition to the range of single hidden networks, a network was trained with three hidden layers, indicated as a red point and "x" mark on the graph. This is a $[2-8-8-2-1]$ network, meaning it contains 2 input nodes and a single output node, and three hidden nodes of size 8,8 and 2. The architecture is the same as the SHL network, with each layer fully connected to the adjacent layers. It is included in this plot for comparison, but the reason for its use is based on the experience with the more complicated surfaces in the other test cases. The smooth ridge test surface is easily approximated using a low number of nodes, and the additional complexity of the deeper network has not generated any improvement in the goodness of fit, though the error is still within the range of the other results and with an objectively small value.

Test case 2: Ridge with sinusoidal mean

This surface contains a sinusoidal variation in the mean of the Gaussian used to construct it, leading to a "squiggly" ridge which is more complicated than the smooth ridge. The same argument regarding the ability of two $tanh$ neurons to approximate a smooth Gaussian ridge points to the need for a much higher number of nodes if the variability of this surface is to be accurately represented. Therefore the SHL network is expected to yield a poor approximation of the surface. This is confirmed by inspecting the result of the neural network training for several hidden layer sizes. Figure 5.31 shows the result of training a SHL network with 10 nodes. It is evident that the points generated by the neural network do not capture the variability of the surface in the x_2 direction. The neural network output

resembles the smooth Gaussian ridge of the previous test case, which may be useful if the features causing the sinusoidal variation were due to noise or were otherwise undesirable. This is because to capture each of the smaller ridges and valley, the network has to contain a sufficient number of nodes. The inspection of the surface fitted with a network containing 100 hidden nodes shows that this is not the case, as can be seen in Figure 5.32. The points generated by the SHL network pass through the actual surface with little improvement over the 10-node case.

The experience with the SHL of various sizes for this test case points to the need for either a very large network, or an alternative approach. Based on the fact that the network reconstructs features by combining the outputs from different neurons, a deeper network with three hidden layers was investigated. The choice was made based on the ability of this network to approximate the test case surfaces very well, as well as the insight that a single layer approximates the average response. In principle, the additional layers operate on the output from the first hidden layer, which has extracted the "smooth ridge" part of the response, and each additional layer operates on components from that first processing of the input data. Another reason for this choice was the observation in several of the test cases that splitting a network of a certain size into a network with fewer neurons but more layers had a markedly increased ability to capture complex shapes in the test surface. This behavior occurs because of the complex features generated by the multiplication of $tanh$ responses from earlier neurons in consecutive layers.

The comparison of the performance of the SHL neural network of different sizes shows a similar trend as that of the first test case, though with a much larger error. In the case of the sinuously varying ridge, there was never a case where the SHL outperformed the MLP network with additional hidden layers. Additionally, increasing the SHL network size from 10 hidden nodes to 100 hidden nodes did not lead to obvious overfitting as in the case of the smooth ridge, indicating that a much higher number of neurons is perhaps needed to obtain adequate goodness of fit using the single layer network.

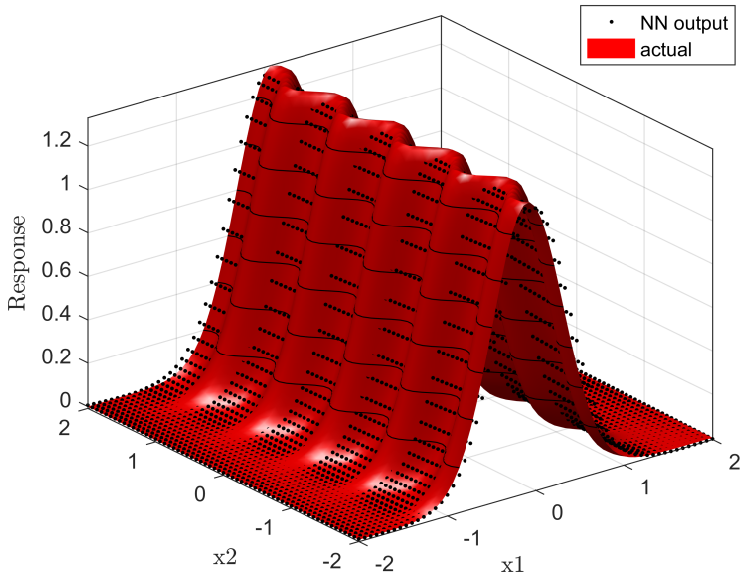


Figure 5.31: Sinusoidal ridge with 10 node SHL

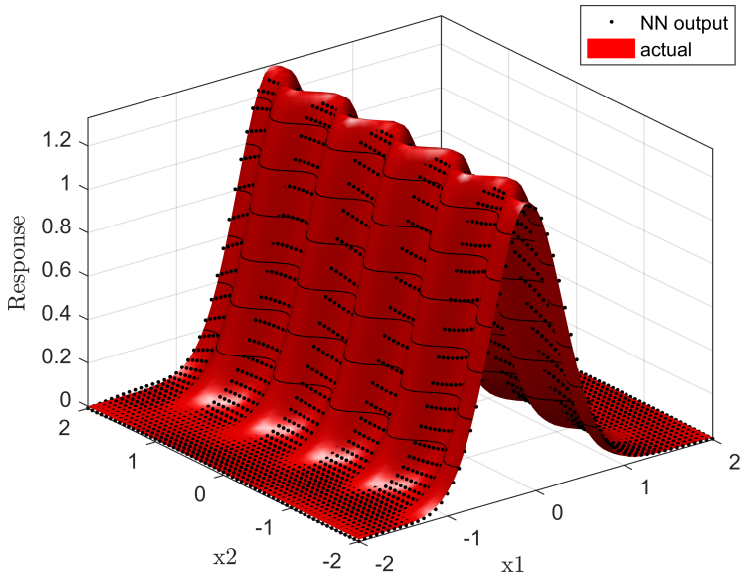


Figure 5.32: Sinusoidal ridge with 100 node SHL

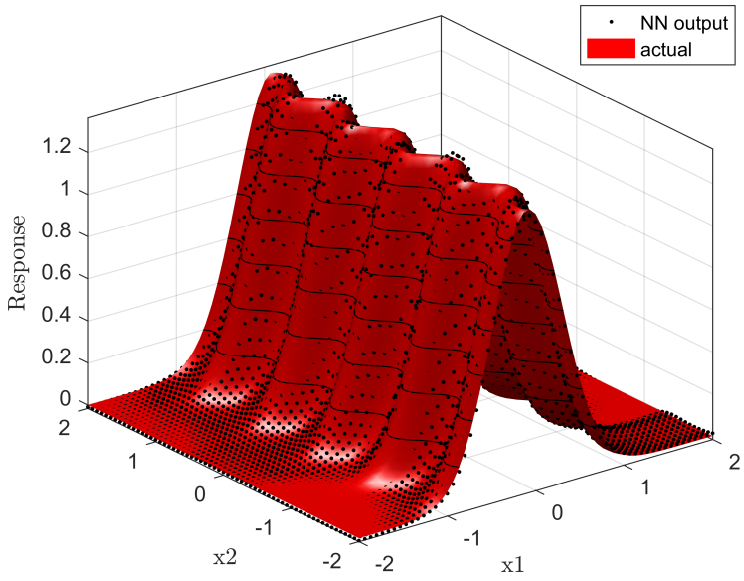


Figure 5.33: Sinusoidal ridge with [2-8-8-2-1] MLP

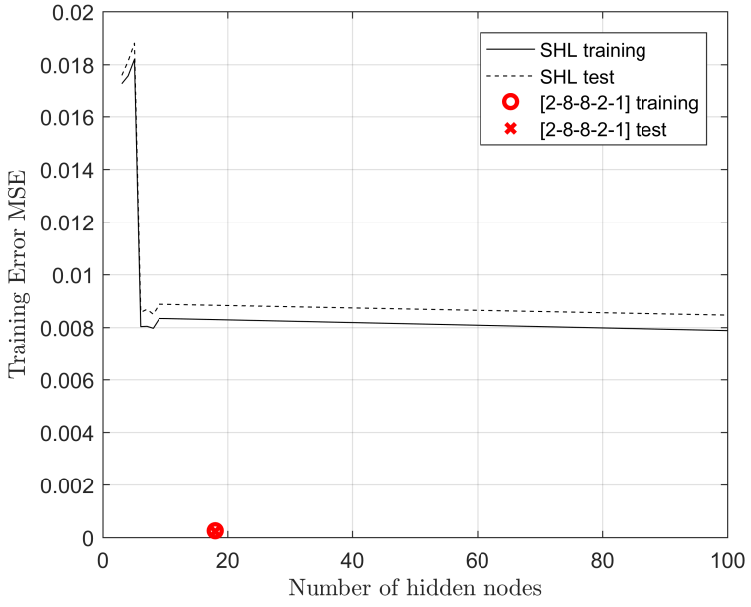


Figure 5.34: Sinusoidal ridge - MSE of different neural networks

Test case 3 and 4: Gaussian peak and truncated ridge

The test cases were constructed to approximate the types of surfaces observed as output from the 2-D dynamic simulation, and are contain many of the features observed in the initial results. One of these features is the truncation of the surface at the maximum value of time experienced during the simulation run. This is an arbitrary truncation and several approaches were considered, such as domain segmentation and smoothing. The flexibility of the present approach, however, allows this type of surface to be adequately captured using a neural network with a few hidden layers of a moderate number of nodes. Another feature seen in initial simulation results is the possibility that an output surface may be multi-modal, which is approximated by the surface with two Gaussian peaks. If the MLP network can fit these types of surfaces, then it can be reasonably expected that the final output from the simulation using FLIGHTLAB can be modeled using a similar approach.

The results of the structured training of SHL networks with increasing number of hidden nodes showed a similar result as the other test cases. Using 100 hidden neurons, the approximation of the truncated ridge still fails to capture the sharp edges at the top of the ridge, as shown in Figure 5.35. Further investigation was not performed for even higher number of hidden neurons, though it is certain that at a high enough number the fit will improve. This is potentially troublesome if the number of training points is considered. As the number of hidden nodes in a SHL network begins approaching the number of samples in the training set, the network can "memorize" the training data and fit it perfectly. To avoid overfitting, the number of hidden neurons must be kept comparatively lower, along with ensuring proper validation through testing on unobserved validation samples.

Figures 5.36 and 5.37 show the performance of the networks of various size, with both training error and test error (on a separate validation set) shown on the same graph. In both cases the trend is toward progressively smaller error with increasing numbers of hidden nodes in the SHL, suggesting that a higher number of nodes would yield adequate results in this case. However, in both cases, the [2-8-8-2-1] MLP outperforms the SHL network.

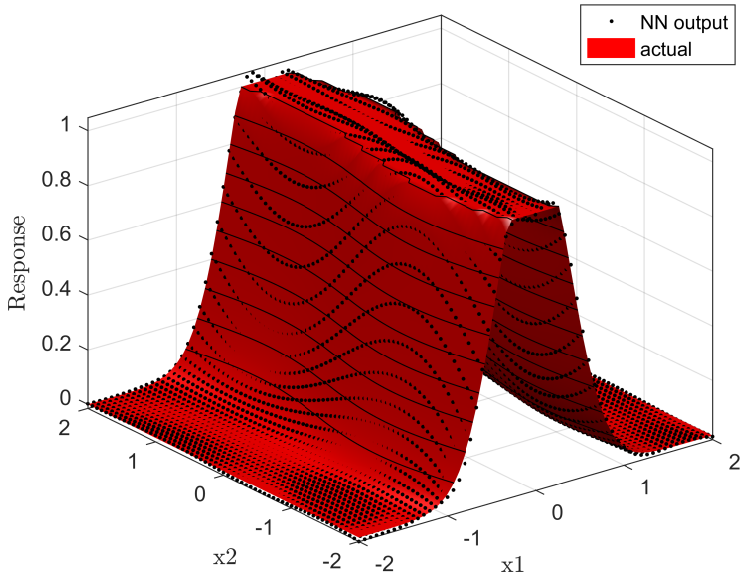


Figure 5.35: Truncated ridge fitted with 100 neuron SHL network

In addition to the error metrics, it is important to ensure that none of the relevant features of the original data are omitted in the model.

Figures 5.38 and 5.39 are fitted with the same [2-8-8-2-1] MLP network architecture that was able to accurately represent the surface in the second test case. It is evident from the figures that the neural network output contains all pertinent features of the data, including the sharp edges of the truncated ridge which are usually very troublesome for other regression approaches and frequently require segmentation of the input domain.

With the analysis shown in this section, it can be concluded that the MLP provide much greater flexibility than the SHL in fitting complicated dataset. The goodness of fit metrics for the 2-8-8-2-1 MLP consistently perform as well or better than the SHL, in some cases by an order of magnitude or more, as can be seen in the additional performance metric results contained in Appendix B.

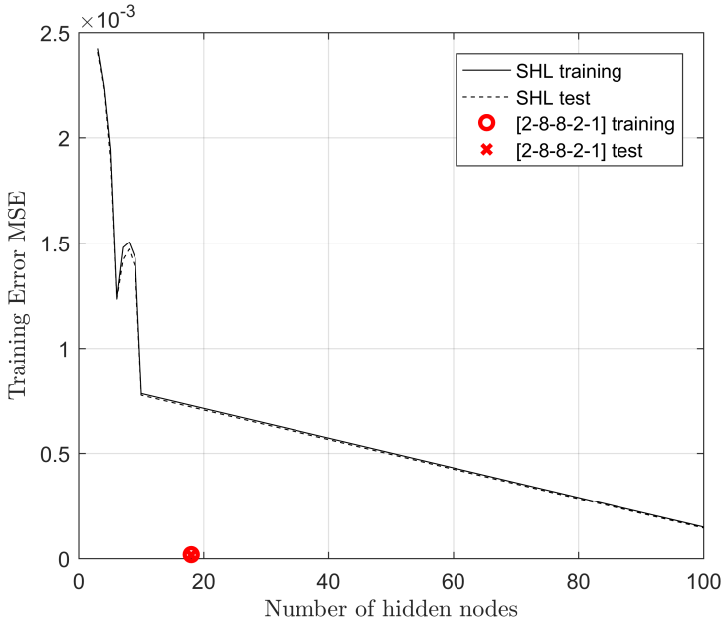


Figure 5.36: MSE - Truncated surface

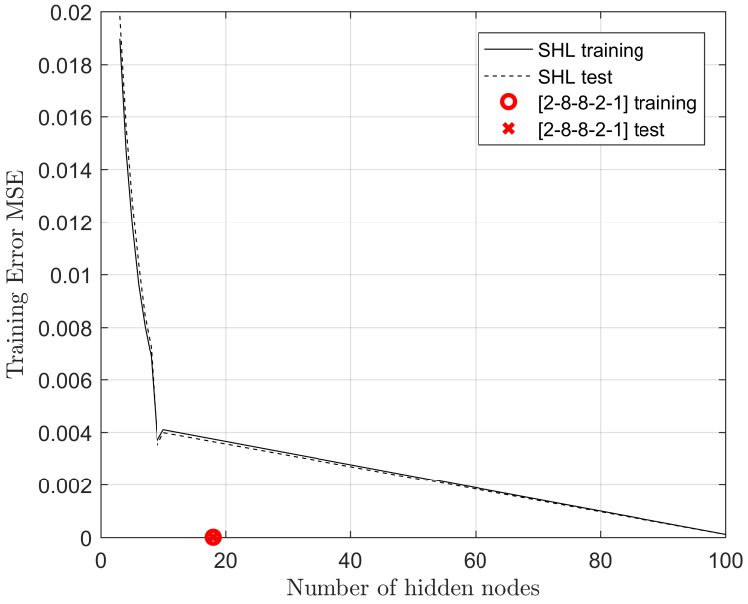


Figure 5.37: MSE - Gaussian peak surface

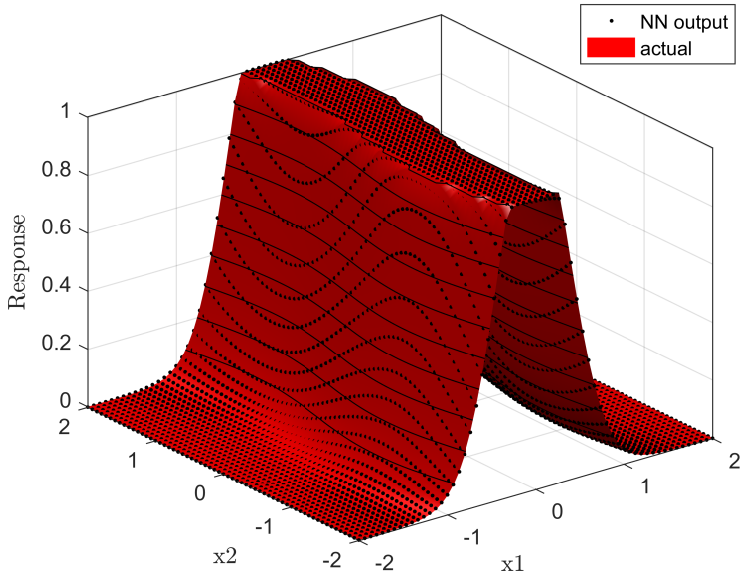


Figure 5.38: Truncated ridge fitted with [2-8-8-2-1] MLP network

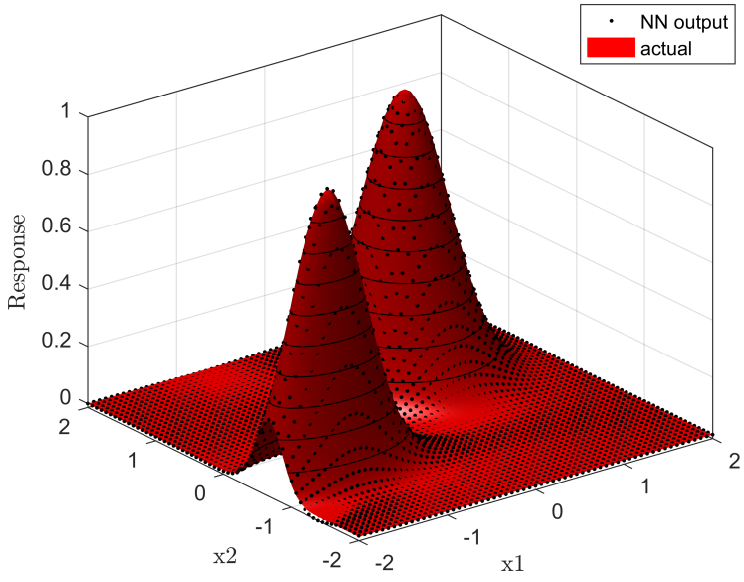


Figure 5.39: Gaussian peaks fitted with [2-8-8-2-1] MLP network

5.4.3 Summary

It is shown through examples and by consideration of the feature construction mechanism in a feedforward neural network such as the MLP, that it is very difficult to know ahead of time what type of network is appropriate for any given data set. The informal guidelines many practitioners adopt regarding the relationship of the number of input variables to the appropriate number of hidden neurons is not a valid criterion to select an appropriate network, except to avoid overfitting. The appropriate network is usually problem-specific, and through the examples in this section it was demonstrated that in addition to the number of inputs, the features present in the data have a marked effect on the number of hidden nodes that yield the best fit. Given the ability of the MLP network to perform with similar error metrics as the best SHL network in the simplest case, and drastically outperform the SHL in all other test cases, it is concluded that a MLP neural network with at least two hidden layers will be needed to fit the final dataset output from simulations with FLIGHTLAB, if these outputs also contain complex features similar to those in the test cases. This analysis gives a partial support to the Hypothesis for research question 5, which will be further supported if the actual FLIGTHLAB simulations can be similarly well represented using a neural networks.

5.5 Test case using FLIGHTLAB

The goal for this methodology is to provide a repeatable approach to determining operational limits based on the physics of the vehicle of interest, so that objective and physically relevant detection thresholds may be utilized in an HFDM system. The reasoning is that if the result will be a much closer approximation to the true operational boundary than the current style of HFDM events, which are frequently defined on a subset of the relevant parameters. This observation is based on a survey of the types of events used in industry, as well as the particular case of autorotation practice examined using a condition-specific model of

the vehicle power in steady-state longitudinal flight. By providing a closer approximation to the true boundary, false alerts can be reduced, especially if additional derived parameters or external data are used in conjunction with the model. Another component of the model-based approach is the use of dynamic models to develop detection thresholds proactively by virtually operating the vehicle until a failure is experienced. This was shown to be possible with a simplified dynamic model of the tipover during taxi, where boundaries were identified and related to the control inputs. If this type of approach is exercised, it may be possible to observe virtual failures that have not yet been experienced in real-life operations, and make appropriate modifications to any monitoring system so that flight operations with increased risk of the simulated failures can be detected. In the present state of practice, these types of events are often missed, usually due to poorly defined or nonexistent detection thresholds. Thus even condition-specific dynamic simulations offer an advantage over the trial-and-error approach to the definition of traditional HFDM events.

The condition-specific models examined in the previous test cases are only valid for use at a narrow set of operational conditions, and consider a subset of the full set of parameters that might affect the operation of the vehicle. While offering an improvement over standard HFDM events, a condition-specific model results in a simplified detection boundary that may miss important effects. Whether these effects are operationally significant or not depends on the condition being investigated, but in the general case a full flight envelope model would be preferred to a condition-specific model. To fulfill this need and to showcase the approach on an specific vehicle, the present experiment examines the condition of tipover during taxi using the nonlinear dynamic simulation FLIGHTLAB.

5.5.1 FLIGHTLAB® helicopter simulation

FLIGHTLAB® is an industry-standard rotorcraft simulator used for both piloted and engineering analyses, and has a component library which allows an investigator to choose a suitable level of fidelity. The rotor rotor is modeled as an articulated rotor with flapping

and lagging degrees of freedom for each blade. The forces and moments due to the main rotor are calculated using the blade-element approach. A three-state inflow model is used for the main rotor, and aerodynamic lookup tables are used for sectional airfoil properties and fuselage aerodynamics, as well as any other aerodynamic surface. The tail rotor is modeled with a uniform inflow, which has been shown to be an adequate model for the low-frequency analyses performed here. The model uses a rigid fuselage, and a simplified engine model. The present work utilized the included UH-60A model which has been extensively validated and utilized directly or as a baseline for a variety of flying qualities, dynamic interface and external load studies. This simulator has been used and validated in numerous research efforts, particularly with regard to the standard UH-60A model provided with the software.

Wan[157] used this software as a "black box" simulation to develop an adaptive controller. Bottasso[102, 158, 159] created a simulation environment that interfaces with external simulators, such as FLIGHTLAB, and performs maneuvers parametrized using the target in a cost function of an optimal control problem. These studies used Model-Predictive Control, where the controller was optimized without direct access to the underlying equations of the flight simulation model.

[160] performed slung load analyses using FLIGHTLAB and introduced several extensions aimed at accommodating the slung load itself. The vehicle model was of the same type used in the present work, and the result The FLIGHTLAB model is one of the industry standard models for analyses where a high level of physical modeling fidelity is required, such as when the vehicle response is to be predicted without extensive use of flight data. The model has been shown to yield good agreement with instrumented tests, especially in the on-axis response. On-axis response is the behavior of a primary degree of freedom in response to the control input that affects it most directly, such as the roll response to a lateral stick input. The off-axis response is not always captured with the same degree of agreement, though extensions such as wake-skew have been shown to improve the corre-

lation. Further model refinement beyond the standard FLIGHLAB model is beyond the scope of the present work, and is a prime candidate for future investigation if high-quality flight data from real-world operations becomes more readily available.

The results of the analysis should clearly indicate the types of operation that would lead to increased risk - and can readily be used in communication with the operators and flight crews for training purposes. To fully realize the benefit of the model-based monitoring scheme, the simulation results would have to be represented within the HFDM system which operates on flight data from real operations. Because of the more complex nature of limit boundaries generated with nonlinear simulation, simple parameter thresholds are usually not adequate for reliable detection. The neural network regression approach to representing the simulation output in a monitoring system is demonstrated here with the results of the nonlinear dynamic simulation of ground taxi operations.

5.5.2 Experimental setup

The final test case is used to test the DEMMoS methodology in a more realistic setting using an industry standard nonlinear flight dynamic model. The steps followed are as described in the development of DEMMoS:

- Step 1: Determine initial condition and define the input space.

  The initial conditions are selected to be in the proximity to the type of behavior that is under investigation. In this study, ground maneuvering was investigated, so the starting condition was a steady ground roll at about 10kts forward speed. The controls were trimmed to maintain this state, and a frozen snapshot of the simulator state was used to begin each simulation run.

- Step 2: Define model.

  This investigation utilized the model provided with the FLIGHTLAB simulator, so this step is complete at the outset. In a more general case, the process would be-

gin with the development or acquisition of an adequate model. The only significant modification to the generic model was the addition of several contact points along the extremities of the fuselage and the rotor. These were in the form of additional landing gears which when compressed would indicate a collision with the ground. A secondary purpose of the additional contact points is to ensure the simulation can continue running after a contact with the ground has occurred and to allow post-processing of the entire data set. A more complete description of the exact implementation can be found in Appendix C.

- Step 3: Generate a range of control input vectors.

  The inputs to the simulation are the pilot's controls, specifically the lateral cyclic, tail rotor pedal and collective inputs. The longitudinal cyclic has little effect on the actual rollover, so was initialized at the trim value and was not used as an input variable for this experiment. The exact values used are in the range $[-30, 30]$ percent for both the lateral cyclic and tail rotor pedal, and $[-10, 5]$ percent for the collective. Because of the highly nonlinear nature of the helicopter response, coupled with the derivation of a time-based risk metric, typical DOE techniques for reducing the number of simulations that need to be run could not be directly applied. Therefore the full set of input variable combinations were analyzed.

- Step 4: Run simulations using the control inputs determined in Step 3.

  MATLAB® was used to define the individual test cases and to pass the necessary inputs to FLIGHTLAB. The lateral cyclic and tail rotor pedal were evaluated in 2 percent increments, whereas the collective was evaluated at 5 percent intervals. The simulation is operated in an open-loop fashion according to the sweep approach, with each control being fixed for the majority of the simulation. An initial half second period is common to all cases, so that all control inputs are applied within the duration of the simulation.

- Step 5: Analyze the resulting trajectories for ground contact and any other limit parameter of interest.

  The resulting trajectories may contain any number of output states selected by the analyst. In the present study, the unintended contact with the ground was the primary consideration, so the additional contact points were the limit parameters of interest. However, other parameters may be easily investigated, such as the vehicle's speed, sideslip, or yaw rate, which are considered to be significant during ground maneuvering. Other simulation outputs, like the moments generated by the rotor, flapping angles, load factor and inflow states are also available and may be useful at other flight conditions. Each trajectory is post-processed to identify the first hitting time for the limit parameter, which is then mapped to the simulation inputs. The result of this step is a multi-dimensional grid with values of the time-to-critical condition criterion associated with each data point. The expected result is that the vehicle remains safe for moderate control inputs, and that an increase in risk/decrease in time to critical conditions is identified for more extreme inputs.

- Step 6: Perform regression of the first hitting time map.

  The results from the previous steps may be used directly by operators, training personnel and HFDM analysts as a source of information regarding the operational scenario investigated using the simulation. HFDM-type events could potentially benefit from having an objective data set available to guide the definition of monitoring boundaries. However, incorporating the data into a monitoring system requires the information generated through the simulation and post-processing effort to be transcribed. This step is accomplished using the MLP neural network with several hidden layers. The network of choice is trained on the full set of multi-dimensional data, and a separate dataset generated using different settings for the control inputs is used to validate the fit. Some trial and error may be required to obtain adequate fits if the first

hitting time response is significantly more complicated than the case used during the development of the MLP regression approach.

At the conclusion of step 6, the trained network would be ready for use in a monitoring environment. The present test case differs from an industrial-scale application in the reduced number of input variables considered and the number of operational conditions analyzed. A successful demonstration of this approach would mean that monitoring levels may be defined in a fairly accurate manner and potentially ahead of any adverse occurrences in actual operation. The ability to directly implement the results of such an analysis may contribute to the improvement of operational safety by allowing more timely and reliable detection of adverse conditions in helicopter flight operations.

5.5.3 Analysis of tipover risk

A summary of the experimental matrix is shown in Table 5.2, which represents more than 10 hours of operational time originating at the same steady forward taxi condition of 10 kts. Following the initial trimming of the vehicle, the trimmed state is used as an input for all consequent simulations, where the vehicle is allowed to operate for a specified length of time in response to the applied control inputs.

Table 5.2: Test cases for tipover simulation

input parameter	range	no of levels
lateral cyclic	[-30, 30]	31
longitudinal cyclic	8	1
collective	[-10,5]	4
pedal	[-30,30]	31
Total experiments		3844
Total flight time		10.68 hours

The model was set at a gross weight of 17000lb for all of the investigations, and the analysis was performed at sea level. Figure 5.40 shows the cyclic and pedal inputs for

the simulations. The same cyclic and pedal inputs were simulated with each of the four different values for the collective input.

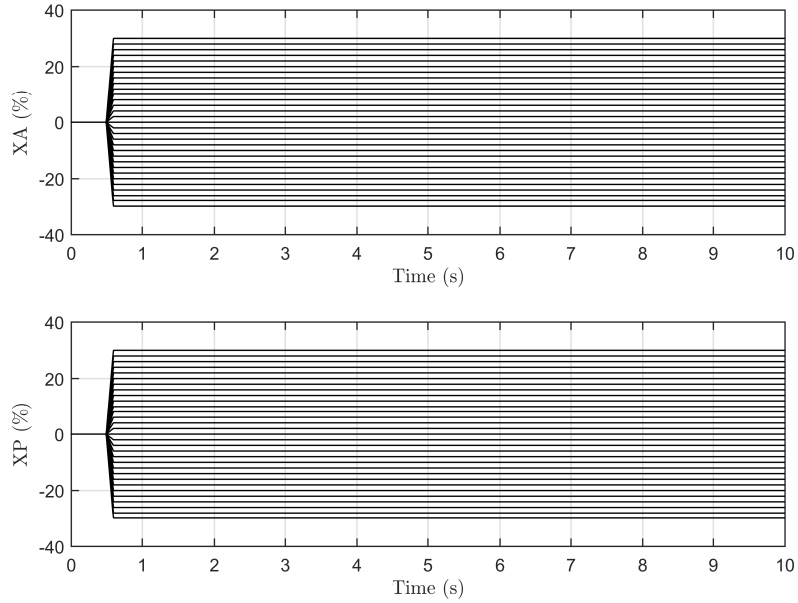


Figure 5.40: Cyclic and pedal control inputs

Figures 5.41 and 5.42 are outputs from the nonlinear simulation. In Figure 5.41 the paths are generated for the lowest collective setting used in this analysis, with the first subset of trajectories showing the normal response. These trajectories are "safe" in the sense than at no point was the ground contacted by the contact points around the perimeter of the rotor. We can see a range of possible outcomes, including steady taxi maintaining the original heading and progressively tighter turns as the control inputs are increased. In this the second part of the figure, the trajectories which experienced a rollover event are shown. The black portion of the trajectories represent the path before the failure occurred, with the post-contact path shown in a light gray line. While the simulation would normally be stopped at this point, the paths are included to show the much greater spread in final positions achieved in these cases. While the post-impact simulation cannot be considered valid, the dispersion is indicative of the increase in velocity when the helicopter tips over without an appropriate reaction from the pilot. This characteristic is the primary driver behind the

dynamic rollover, a particular case of rollover events when the vehicle is constrained on the ground while the main rotor is accelerating the vehicle along the ground.

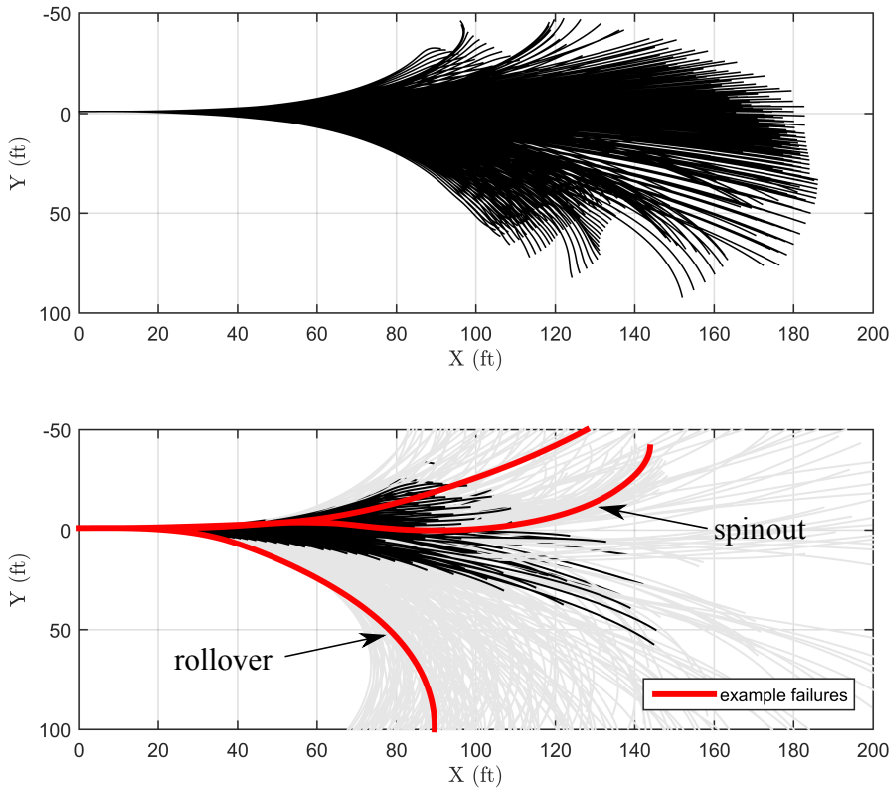


Figure 5.41: Safe and failed ground tracks

The same graphs shows several selected trajectories that are interesting because they show the three subsets of rollover events experienced during the simulation. The upper and lower trajectories marked in red are representative of a typical rollover during ground taxi. Pilots initiate ground taxi maneuvers using a combination of tail rotor and main rotor inputs, where the yaw is primarily the result of tail rotor application. The purpose of using lateral cyclic during ground taxi turns is to cancel the moment generated by the tail rotor about the landing gear contact points with the ground. In cases where inputs are cross-coupled, a rollover is likely to occur, as shown in the plot. This is not the only hazard, however. The trajectory in the center of the figure appears to initially follow the original heading, before

finally heading away. A subset of the trajectories did not immediately develop large roll angles, but instead generated substantial yaw rates that caused the helicopter to spin out. This type of response is a potential hazard during ground taxi, especially with collective settings closer to that required for takeoff.

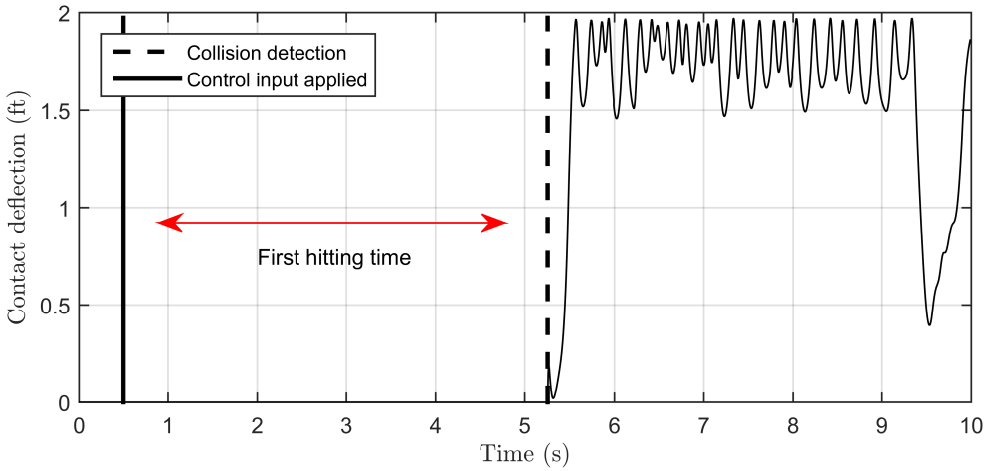


Figure 5.42: Detection of ground collision

Figure 5.42 shows one of the limit parameters used to determine the safety of a trajectory. In this particular instance, a rollover occurred and the contact point is deflected as a result. The vertical dashed line indicates the instant in time when the contact occurred. This detection is performed for all simulation runs and the value is related to the moment of control input application at $t = 0.5s$. This time to a critical condition is then used as a measure of the risk associated with a particular combination of control inputs. The result of performing the entire set of simulation runs is a set of surfaces where the outcome of each run is mapped to the control input grid used in the simulation. Figure 5.43 is one such surface, generated for a collective cyclic setting of -10%. The central part of the surface is flat, corresponding to the control settings which did not result in contact with the ground. At this collective setting, this is a rather large portion of the input space. As more extreme control inputs are applied, the vehicle comes in contact with the ground, with a trend toward faster rollover as the control input is increased. This is represented by the sharp drop-offs in the surface from the nominal value (determined by the simulation run time) to values as

low as 2 seconds for the most extreme inputs. The foreground portion at negative cyclic and tail rotor inputs correspond to a spinout, which results in contact with the ground but with a slight delay. As the collective is increased, it is expected that the region of the control

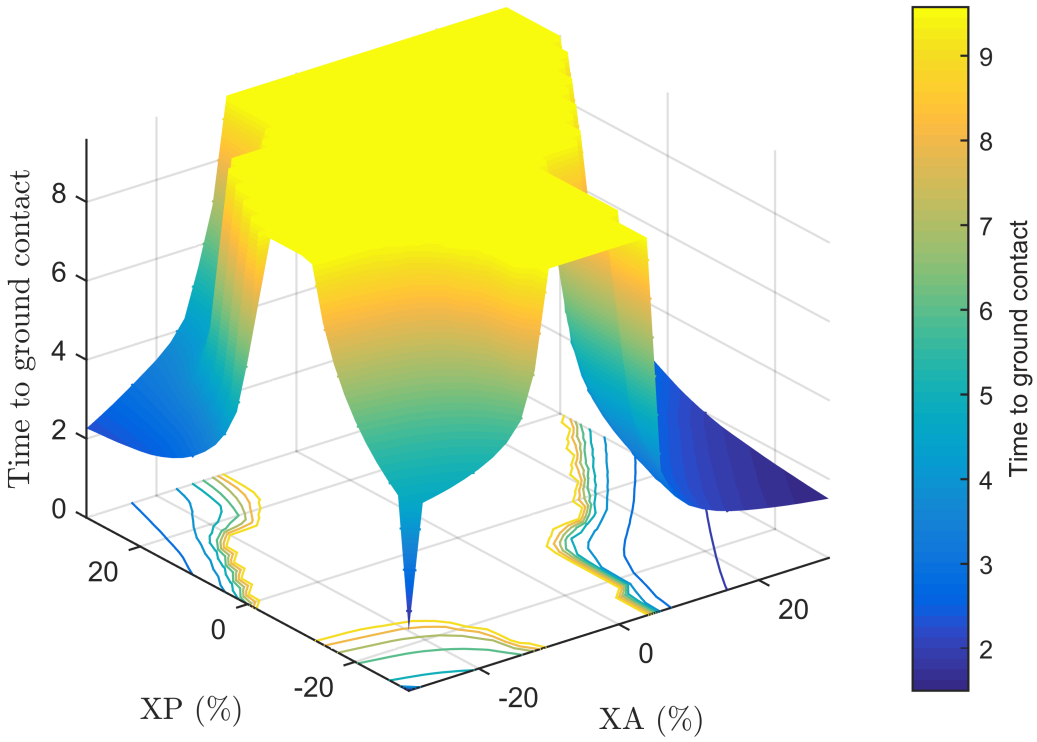


Figure 5.43: Time to ground collision for -10 % collective

input space which results in a rollover is expanded. This is shown in Figure 5.44. The flat portion of the surface corresponding to safe input values has markedly contracted, and the overall response is faster as evident by the lower values of time to contact at the extremes of the control input space. These and two additional surfaces are shown in Appendix A. The access to this type of result would be beneficial to an operator for the analysis of a potential hazard as well as for monitoring purposes. The part of the surface where the drop off occurs from safe to progressively riskier control inputs can be used as a basis for establishing threshold in HFDM safety events. A more complete approach is to collect all simulation outputs, in the form of these input-output maps, and generate an expression which can be

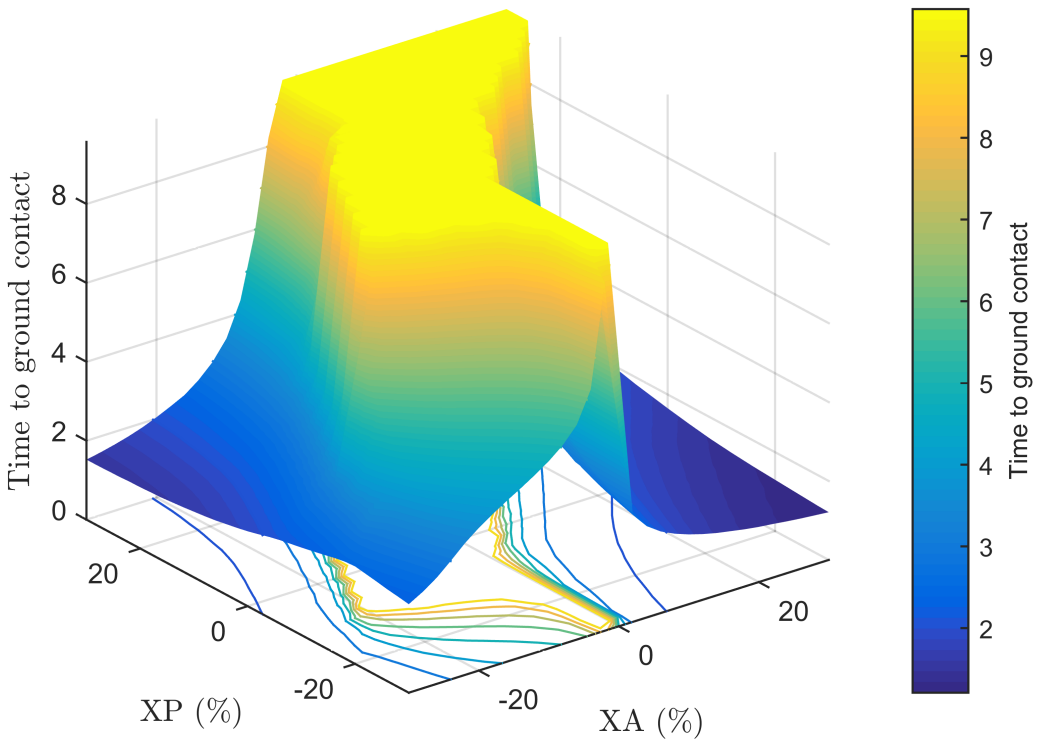


Figure 5.44: Time to ground collision for -5 % collective

evaluated within the monitoring system. An MLP neural network was fitted to the output data shown in this section. The similarity between the test cases used in the development regression approach using the MLP and the current results ensured adequate goodness of fit metrics.

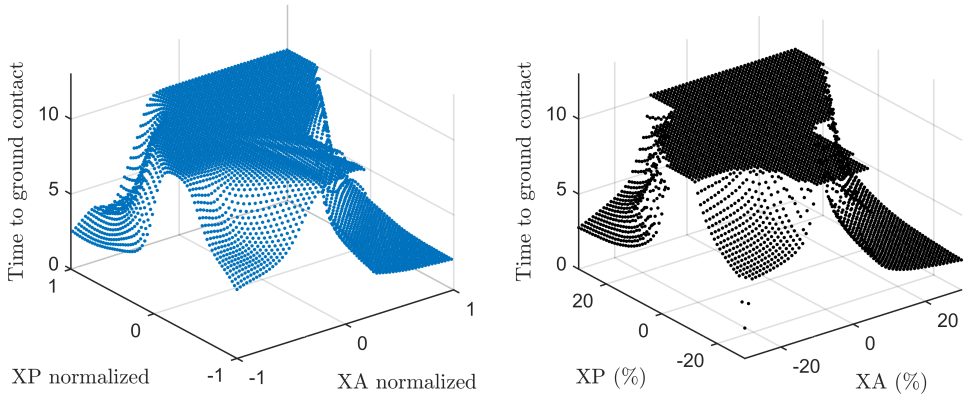


Figure 5.45: 2-8-8-2-1 MLP fit of -10 % collective surface

Figure 5.45 shows reasonable agreement between the original surface on the right and the modeled surface on the left. All pertinent features are captured, including the flat top and the three regions with increasing risk of rollover. The mean of the error between the two surfaces across all settings for the collective is close to zero, at -2.7e-5. The standard error is within 3% though the presence of isolated outliers is a potential concern. This network is not the final solution in the sense that it is optimized but it is a clear demonstration of the potential to readily implemented complex model-based metrics in a monitoring system with computational requirements similar to what is currently used in practice. Using the method demonstrated in this experiment, an operator or HFDM system analyst could identify a condition of interest and perform the requisite simulation to yield a dataset such as the first hitting time metric. These results can then be directly implemented in a database or used for communicating information regarding safety hazards or training.

5.5.4 Summary

This methodology used in this work and this experiment are a showcase of an approach to use physics-based models in the context of HFDM. Therefore this final experiment is intended to demonstrate the process that begins with a vehicle model and knowledge about the region of interest where the investigation would be performed, and results in a set of analysis results that can be readily implemented in a monitoring environment. The simulation is expected to detect critical conditions, and determine where in the input space these conditions occur. Using the time-based metric, risk can then be represented as the time required to reach the critical condition. Finally, the ability to represent the entire set of first-hitting-time results, as opposed to a single level curve at a pre-determined time to a critical condition means that the operator will have the flexibility to determine when their operation should trigger an alert. The operators would thus retain the ability to fine tune the detection thresholds based on their own tolerance for risk or the nature of the operation. At the same time, the analysis done ahead of time would enable the direct comparison of flight data coming from different operators, which has not been the case when operators have different HFDM events from other operators. This aspect of the model-based approach is especially important for large-scale monitoring systems where data from many different operators are stored and analyzed.

CHAPTER 6
CONCLUSIONS AND RECOMMENDATIONS

6.1 Conclusions

Rotorcraft possess unique flight capabilities which make them well suited for a variety of challenging missions, including operations in hazardous conditions and under significant time constraints. Unfortunately, the safety record of helicopter operations is an order of magnitude worse than that of commercial airlines, which has prompted the rotorcraft industry to investigate strategies for safety improvement. Among the safety approaches taken in industry, Helicopter Flight Data Monitoring (HFDM) stands out as an effective means of identifying and mitigating emerging safety concerns before the trend results in an accident. Unlike traditional post-accident analyses of flight data, HFDM and similar monitoring systems operate by accessing, analyzing and acting on flight data recorded during regular flight operations. By comparing flight data to a set of condition indicators, known as safety events or exceedances, HFDM systems have demonstrated the ability to alert operators of unsafe occurrences, and provide objective data in support of mitigation strategies. However, when safety events are poorly defined or based on incomplete knowledge, the ability of HFDM systems to detect safety hazards is diminished, with reduced potential to help prevent avoidable accidents. As a consequence, the condition indicators used in HFDM require continuous refinement to achieve adequate detection performance. Additionally, the detection of any adverse condition can only be achieved if pertinent safety events have been defined beforehand. Mitigating this dependence on pre-defined notions of deteriorating safety has been the focus of several recent studies on the use of flight data for safety enhancements in aviation.

The first question answered by this work was what approaches are available to cre-

ate this enhancement. The answer to this question was obtained through a survey of the available literature, with a general categorization into purely data-driven and model-driven methods. Of these the model-driven methods were preferred given the reduced burden on collecting and analyzing data. In addition, the addition of physics-based models is beneficial for the statistical approaches to enhance the analysis by providing relevant features on which the statistical approach could operate.

The second question that followed immediately from this first choice was intended to identify the manner in which a physics-based model can be obtained or created given the needs and constraints of their intended use in HFDM. Based on a review of types of conditions which are relevant from a safety standpoint, the two primary approaches used in literature are system identification and modeling from first-principles. The second method was selected, again due to the reduced reliance on high-quality flight data and the greater predictive ability if the models are defined appropriately.

This led to the formulation of the DEMMoS methodology, and the remainder of the work was concerned with identifying the components of the method along with demonstrating it on a set of relevant safety hazards. Since the physics-based modeling approach was chosen, a natural next question was to identify the types of effects and components of the model which would provide insight into the conditions of interest. This led to the hypothesis for RQ3 which stated that if the model contains the relevant effects, its application within HFDM would enable the definition of safety events with improved detection performance. It was found through experimentation that even fairly simple models can approximate the limit boundaries for narrowly defined conditions, and improve the detection performance over the existing detection thresholds, thus offering support for the hypothesis associated with RQ3. The DEMMoS method using static models was demonstrated using simulated and actual flight data in the detection of VRS and for autorotation monitoring. The primary result, in the case of autorotation monitoring with actual flight data, is that false alerts and missed detections were mitigated through improved correlation between

the modified event boundary and the instantaneous flight condition. The method was also demonstrated with a dynamic model on a ground maneuver that has produced accidents in the past. Both 2-D and 3-D models were used, leading to the insight that despite the utility of simpler models, it is desirable to retain as much of the mathematical formulation of the vehicle physics for all investigations. Modeling with a more general dynamic model using the dynamic sweep propagation method revealed the ability to identify boundaries through time-domain simulation, yielding both a value for risk in terms of time to reach the critical condition as well as its location in the input domain.

In addition to the physics-based modeling, the experiments included an investigation of several approaches to manipulate the simulation in cases where dynamic simulation is required. Several approaches were identified based on a survey of the literature and it was hypothesized that the chosen alternatives can elicit the results of a baseline optimal approach without pre-specifying the form of the response. This hypothesis is supported with certain caveats. Specifically, the randomized control input approach was shown to explore the operational space, visiting every possible trajectory as the number of trajectories increase. However, given the need to map these results to the input conditions and simulation variables, it was found necessary to use a fixed-input control scheme, which can be thought of as a single segment from the alternatives investigated during the experiments. When iterated over a range of initial conditions, the discrete sweep allows the operational space to be investigated while preserving the input-output mapping. Iteration is necessary to build up a set of training data for the final component of DEMMoS, the mathematical representation of the simulation outputs in order to enable monitoring within a typical HFDM database. This investigation provided an answer to the methodological concern of how the simulation should be propagated to elicit a response without a priori definitions.

The fact that DEMMoS can be executed with static and dynamic models leads to two different manners in which the outputs are processed. In the case of static and condition specific models, the output is designed to correspond to the desired detection scheme. In

case of dynamic simulation, and given the complex nature of the helicopter response and the similarly nonlinear risk metric results, it was essential to develop an efficient way of implementing the results in a monitoring framework. It was hypothesized that neural networks of the feedforward type with multiple hidden layers would perform well given the insight into the type of simulation results that can be obtained given a dynamic helicopter model and a time-based safety criterion. An investigation was performed which offered support to this hypothesis by obtaining acceptable fits using the proposed neural network architecture. Specifically, a multiple-hidden-layer network was identified as an acceptable general model to represent the results in a form which is conducive to implementation in a monitoring database. After sufficient simulations are completed, this approach would allow online monitoring of hazards which have been identified in simulation with the potential to detect adverse flight conditions before they are observed in practice and thus enhancing operational safety.

Finally, the DEMMoS method was demonstrated using the industry-standard simulation FLIGHTLAB® for the same condition of a helicopter maneuvering on the ground. After 10+hrs of simulated operation on the ground, regions of the input domain were identified where the expected ground rollover occurs. Furthermore, a secondary failure mechanism was observed in the form of a spinout, which was not specified as an a priori target. The time-domain results were post-processed to relate the time to any of the ground-contact type failures to the input domain, and a regression was performed using the identified model from the last experiment. The approach generated adequate fits, which capture the complicated form of the response and allow evaluation at interior points of the input domain, allowing monitoring of flight conditions contained in the simulation inputs. The method is envisioned as a support module to a HFDM system in that the simulation component would be iterated to provide data for conditions with known associated hazards or where operational data is scarce, eventually covering a large portion of the operational space.

Overall, the ability to define monitoring metrics by observing virtual failures and the

improved detection enabled through condition-specific models provides an answer to the primary objective of generating improvement over the current state of practice. The results provided are not static, but can be generated for a range of conditions while enabling the operator to set their individual alert levels. Below are some conclusions and recommendations based on the experience in developing this method. Conclusions:

- Model-based metrics designed to track safety conditions can improve the performance of HFDM systems
- Dynamic simulation can be utilized in a structured approach to develop boundaries on flight parameters for safety monitoring
- Unified representation of risk in terms of time to reach a critical (or undesirable) condition
- Output is generated for a range of parameters that may be of interest, and the entire spectrum of results from safe to progressively riskier operation can be obtained
- Reproduced known results and have a methodology in place to perform investigations throughout the flight envelope
- There is a clear path to extend the method to a more "exploratory" approach provided sufficient computational resources are available.

6.2 Recommendations and future work

- Parallel computation

  The analysis was limited by the number of cases that can be investigated due to the lack of a parallel computational capability on the host server. To enable parallel execution of this methodology, specifically of the simulation component, an executable

version of the flight simulation would have to be generated and then executed as separate processes. The present implementation uses FLIGHTLAB, which has the functionality to generate a standalone executable, though special licensing is required. Additionally, the simulation is executed to generate a number of independent trajectories, which could all be executed in parallel. Each trajectory is assembled from segments simulated for a subset of the overall time horizon, and these would have to be executed sequentially. The number of individual simulations would rise with the number of segments in a trajectory, but the duration required to simulate each segment would be correspondingly reduced.

- Simulation models

  The models used in this thesis were wide-ranging in their manner of execution and purpose. The simpler models were used to illustrate the benefits of using a model-based approach, and should be utilized in an HFDM setting. The dynamic models contain the same physical models as the steady state models, though allow the computation of the temporal component which is useful as a unified risk metric.

- System identification revisited

  The models used in this thesis were built up component-wise, which places a burden on the analyst to manually build and validate models. The results should provide a motivation to enhance recording capability across helicopter fleets, which would enable automatic identification of model parameters and validation of the resulting models. Automating this process would provide an advantage to the method presented in this thesis by improving the modeling capability. This would be especially useful in capturing the inherent variation of operating rotorcraft at various sizes and in different missions that may be partly to blame for the detection problems cited in HFDM literature.

- Neural Network optimization

The selection of an optimal neural network configuration was beyond the scope of the present work, but it is a large area of research with promising results to date. The most common approaches to optimizing (though optimum can hardly be guaranteed) is to start with excessively large networks that are pruned, or use a genetic algorithm to evaluate random mutations of a candidate population. Both of these approaches would help automate the process of transcribing the simulation results for implementation purposes.

- Investigate additional known hazards for specific rotorcraft

  This work investigated generally known hazards as well as a tipover hazard which is experienced by helicopters with wheeled landing gear and a high-mounted tail rotor. Operational experience has shown that some types of helicopters exhibit peculiar flight characteristics which may have been inadequately addressed by standard guidance materials. A well-known example is the case of mast bumping on helicopters with teetering rotor designs, which has been experienced in significant numbers on Robinson helicopters. The company flight manuals list several Safety Notices (SNs) that directly address this occurrence and advise pilots to avoid low-G maneuvering, flight into turbulent air and low rotor RPM. Proper recovery techniques are now part of pilot training, but the cause and conditions that could trigger a mast bumping event are not being monitored extensively, and in most cases the post-accident cause of a mast-bump is listed as ”undetermined”.

  A recent article titled *Undetermined Reasons*[161] reiterates the continued need for improved understanding, monitoring and mitigation of the mast-bumping phenomenon, as evidenced by the occurrence of several accidents of this type in recent years, and on additional helicopter models.

Appendices

APPENDIX A

FLIGHTLAB OUTPUTS

The demonstration using FLIGHTLAB used a range of simulation inputs each with a corresponding trajectory. The trajectories were post-processed to determine the presence of a critical condition, which in this case was contact with the ground with parts of the helicopter other than the landing gear, and a relationship between the simulation inputs and the simulation outcome was established. Thus a value of the time to come in contact with the ground was established for every input vector. Figures A.1,A.2, A.3 and A.4 show the mappings between the collective and pedal inputs and the time to come in contact with the ground starting from the straight and level taxi condition.

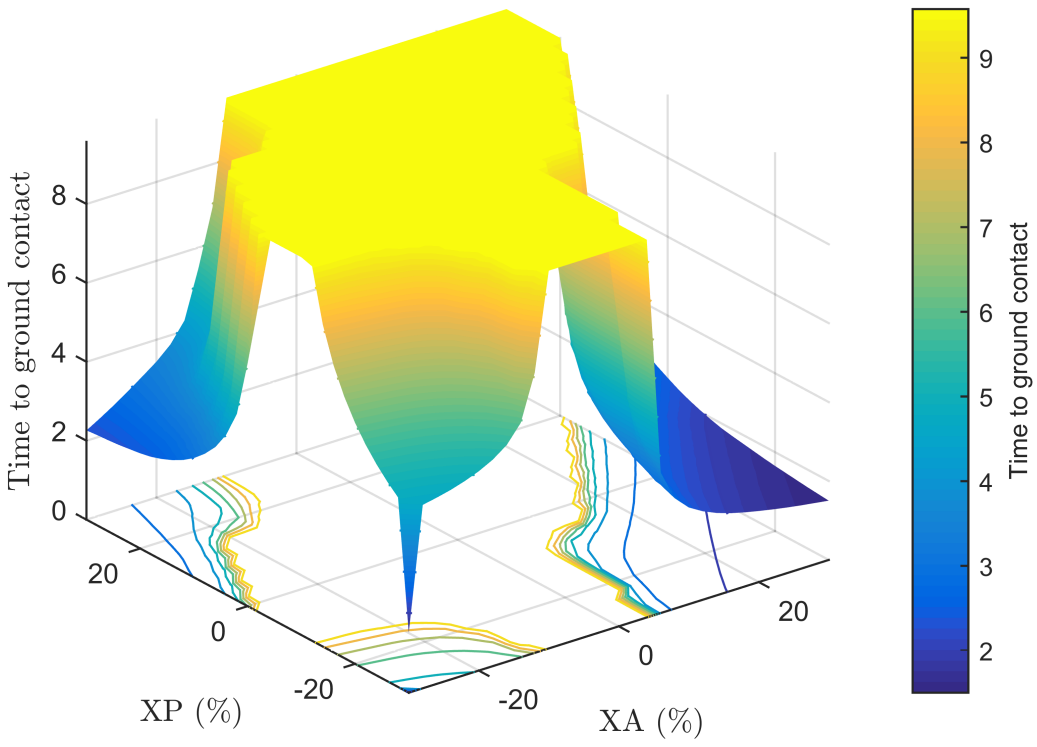


Figure A.1: Time to ground collision for -10 % collective

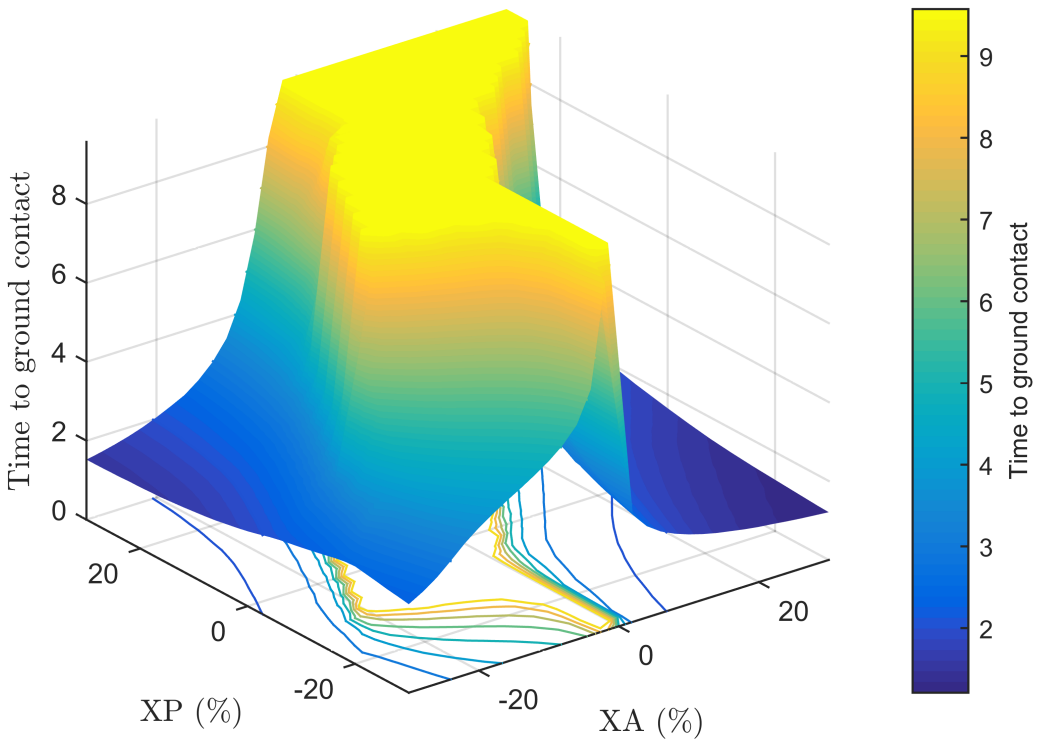


Figure A.2: Time to ground collision for -5 % collective

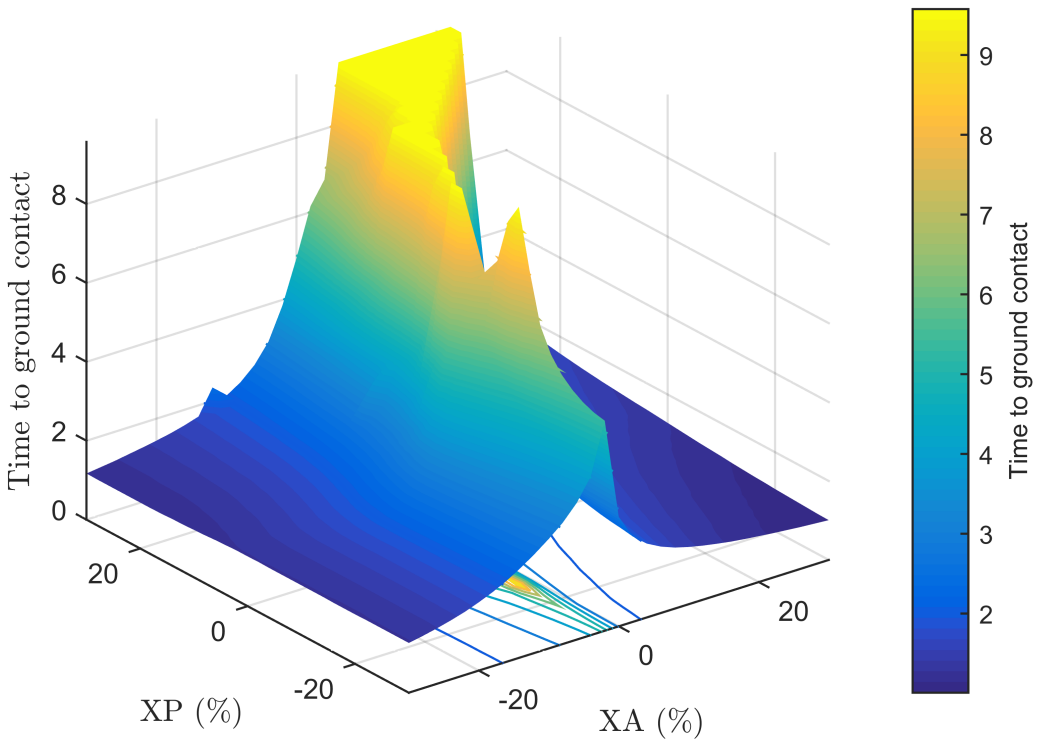


Figure A.3: Time to ground collision for nominal collective

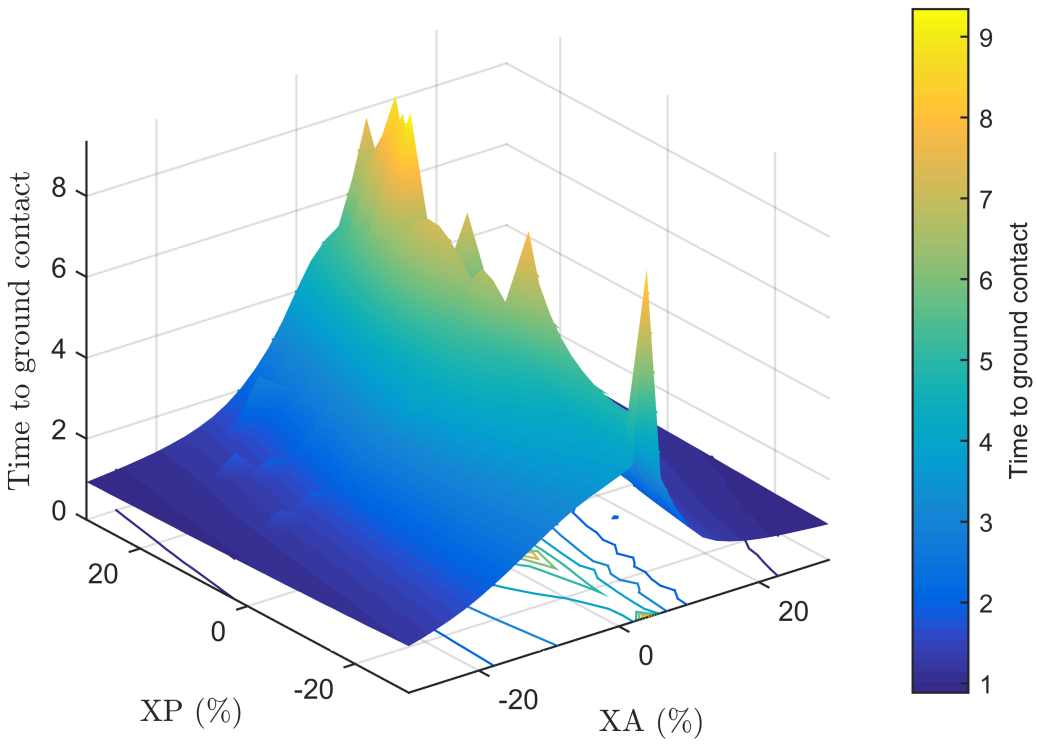


Figure A.4: Time to ground collision for 5 % collective

The collective is changed between each of the mappings, and it can be seen that the region where the end of the simulation time horizon is reached shrinks as the collective control input is increased. This type of analysis is shown for a small number of conditions here, and is expected to be expanded over time, enlarging the region where the detection metric is defined. The detection metric is based on comparing incoming flight data with the output from the neural-network representation of all such mappings for a given vehicle and a range of flight conditions, an example of which is shown in Figures A.5 to A.8 below. On the right side of each figure is the actual data obtained from simulation and post processing of the flight trajectories, and on the left is the representation of the same data using a neural network with multiple hidden layers. In each case, the network captures the complicated nonlinear features of the actual response, allowing the results to be implemented in a monitoring database where quick comparison between incoming data and the simulation output is required.

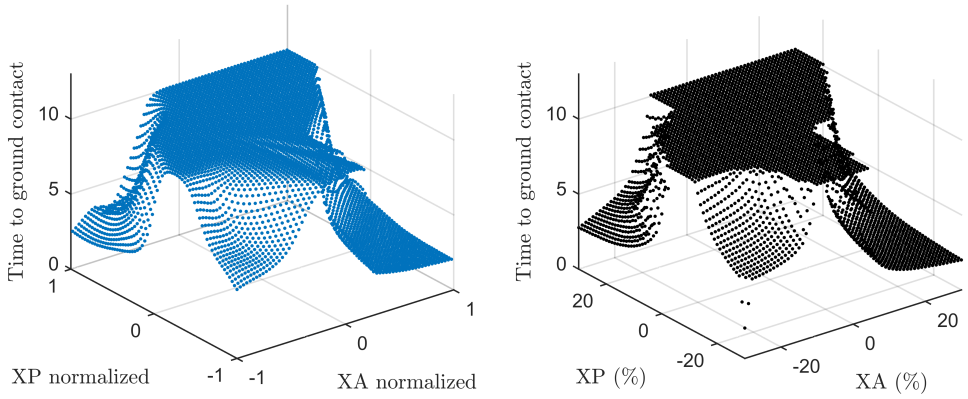


Figure A.5: 2-8-8-2-1 MLP fit of -10 % collective surface

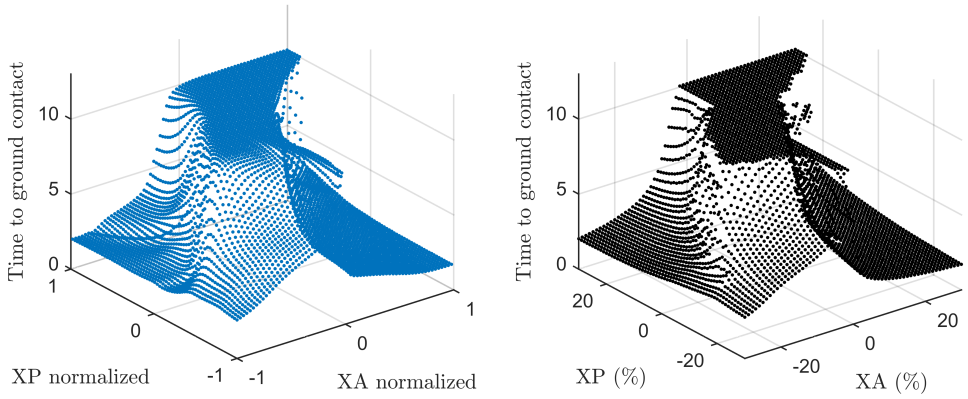


Figure A.6: 2-8-8-2-1 MLP fit of -5 % collective surface

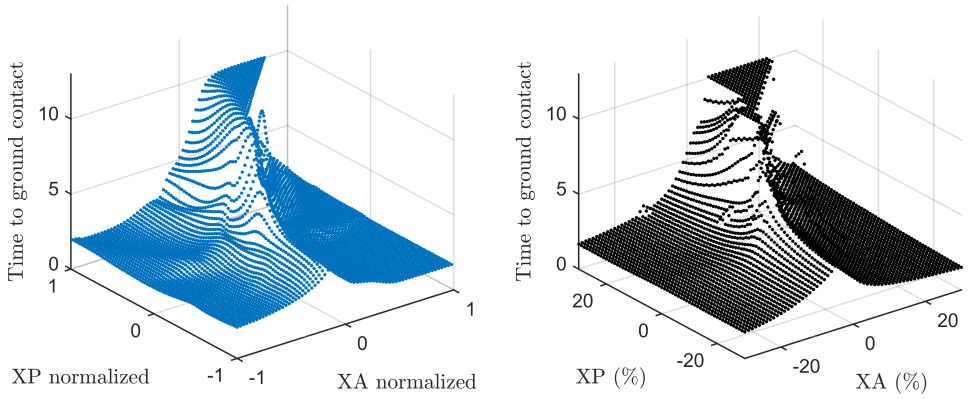


Figure A.7: 2-8-8-2-1 MLP fit of nominal collective response

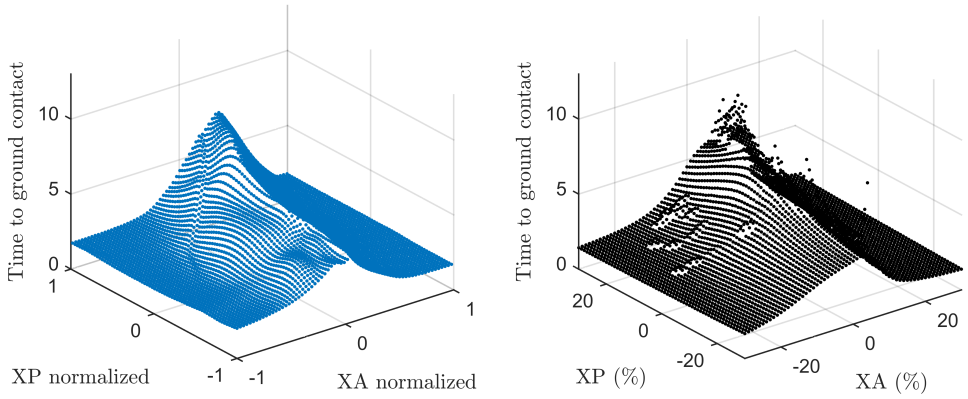


Figure A.8: 2-8-8-2-1 MLP fit of 5 % collective surface

APPENDIX B

NN TEST CASE PERFORMANCE RESULTS

The performance of any surrogate modeling or regression technique is measured by the ability of the fitted model to approximate the original data as well as generate new data outside the data used to create the model. This is commonly referred to model fit error (MFE) and model representation error (MRE) and is calculated as the average squared error between the training and test data, respectively, and the values generated by the model at the corresponding input settings.

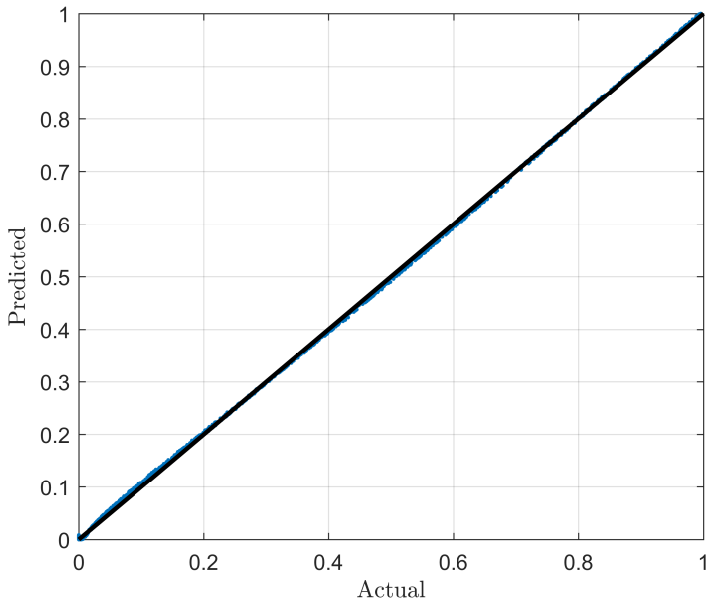


Figure B.1: Actual by predicted - smooth ridge (case 1)

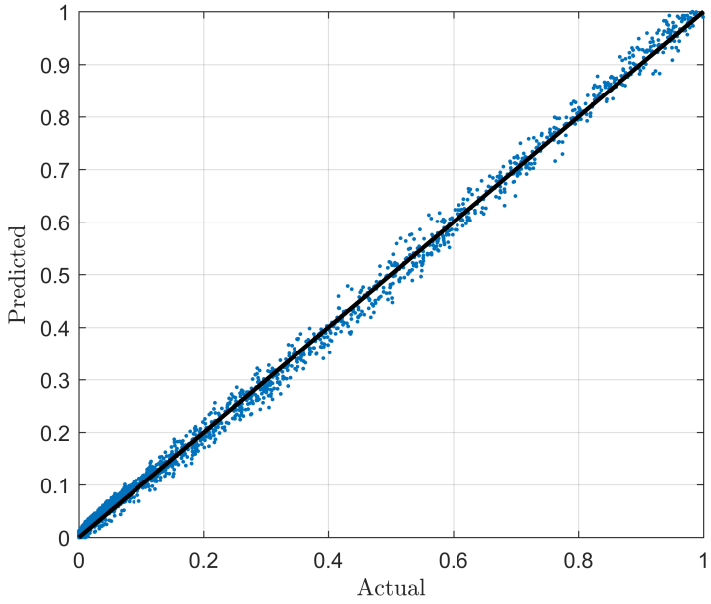


Figure B.2: Actual by predicted - sinusoidal ridge ridge (case 2)

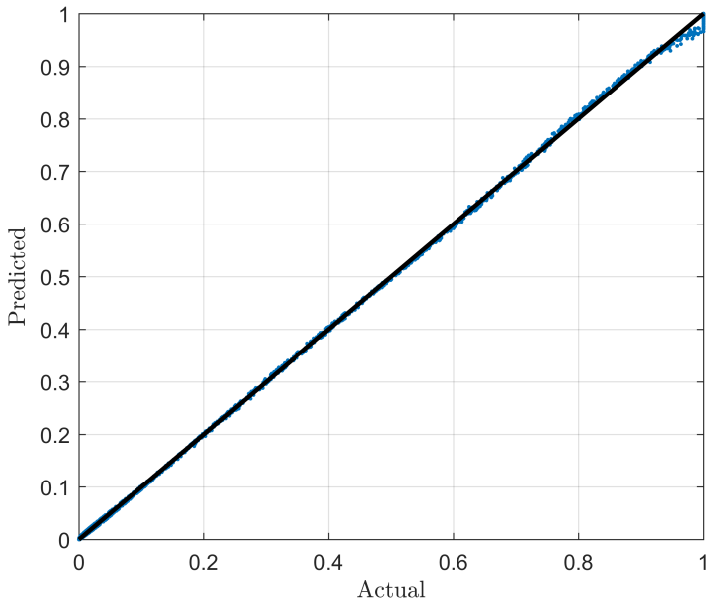


Figure B.3: Actual by predicted - truncated ridge ridge (case 3)

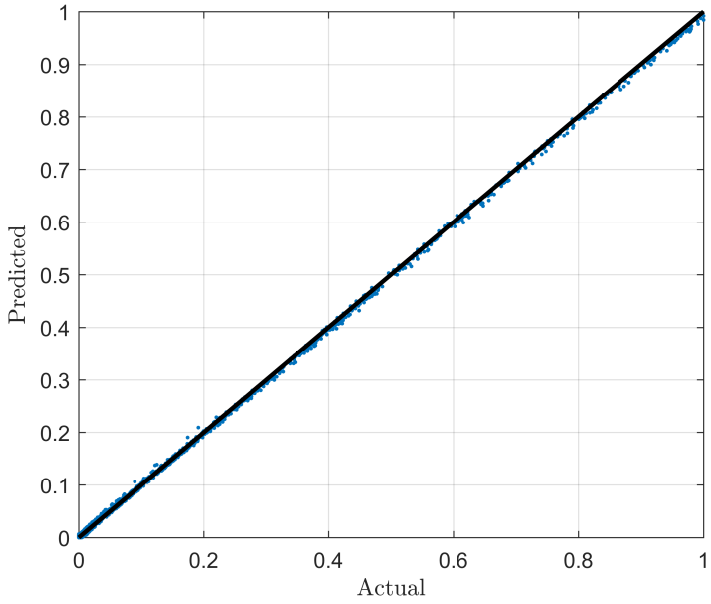


Figure B.4: Actual by predicted - Gaussian peaks (case 4)

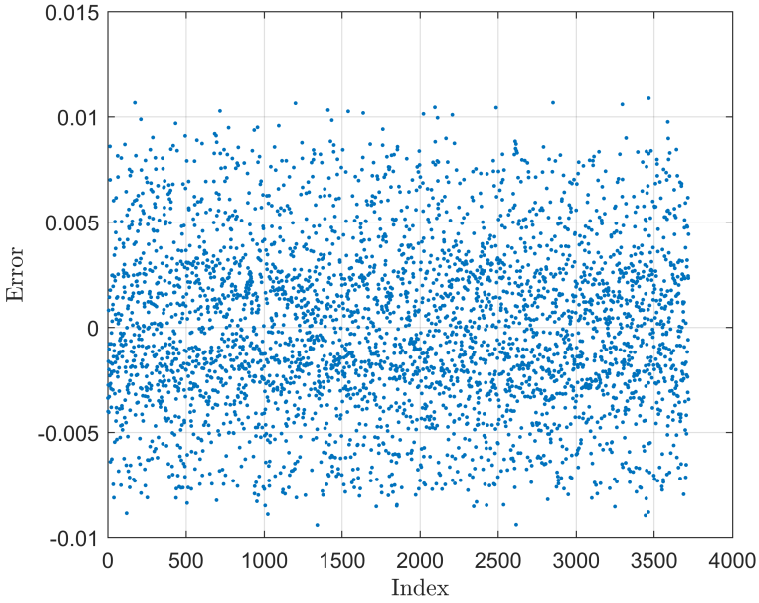


Figure B.5: Error residuals - smooth ridge (case 1)

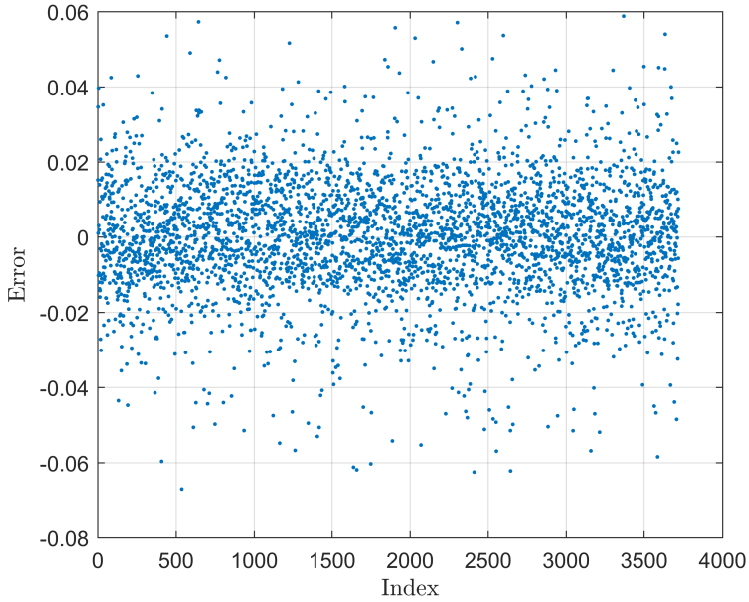


Figure B.6: Error residuals - sinusoidal ridge ridge (case 2)

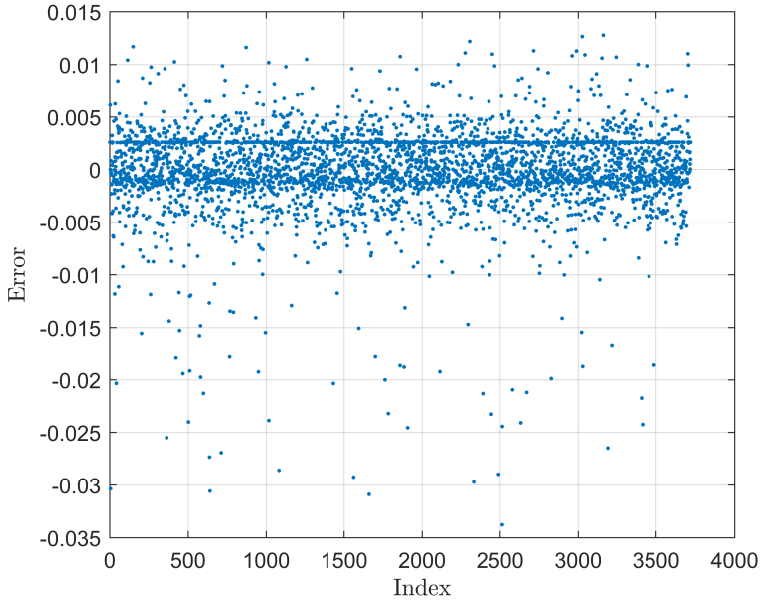


Figure B.7: Error residuals - truncated ridge ridge (case 3)

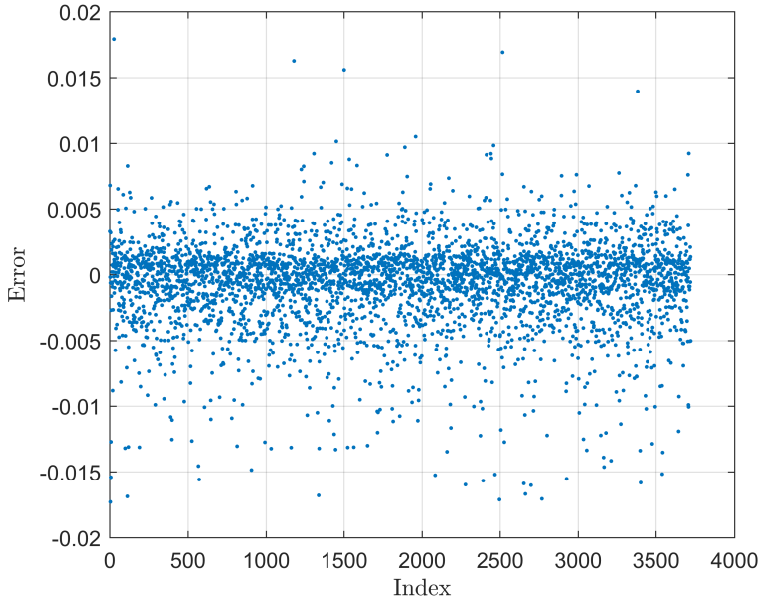


Figure B.8: Error residuals - Gaussian peaks (case 4)

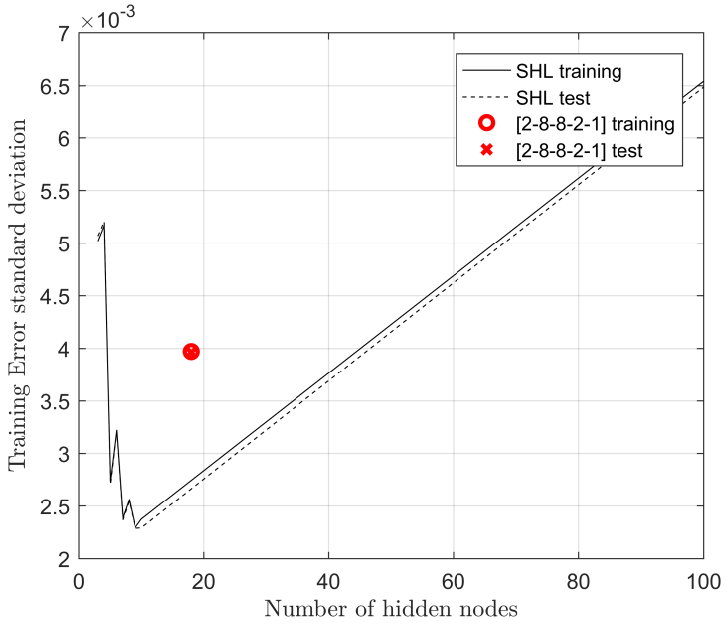


Figure B.9: Standard error - smooth ridge (case 1)

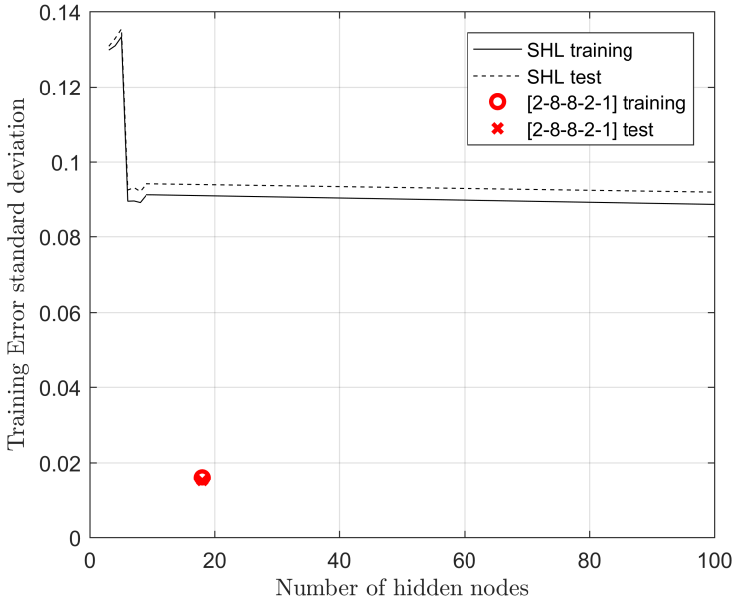


Figure B.10: Standard error - sinusoidal ridge ridge (case 2)

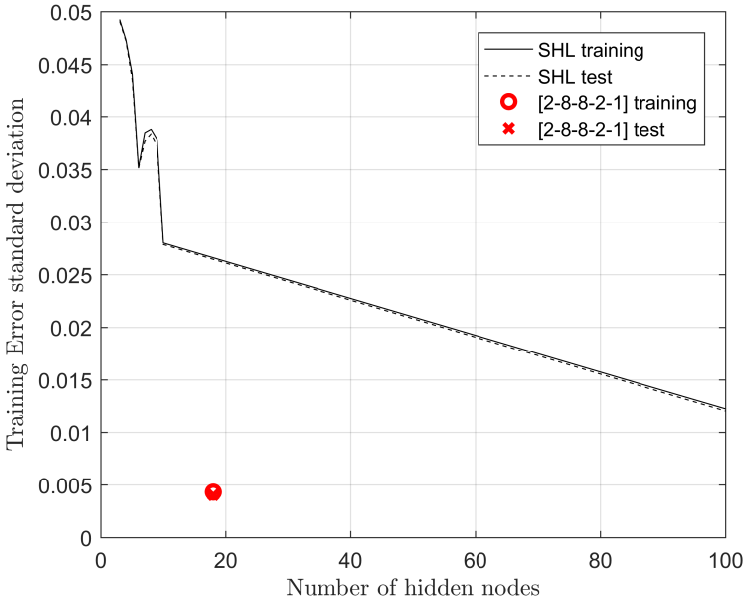


Figure B.11: Standard error - truncated ridge ridge (case 3)

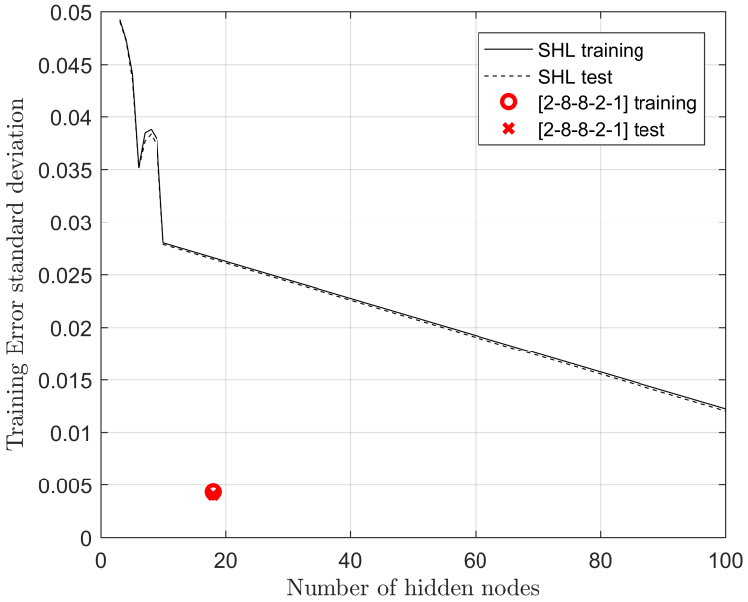


Figure B.12: Standard error - Gaussian peaks (case 4)

APPENDIX C
FLIGHTLAB® IMPLEMENTATION

C.1 Flightlab simulator

The final demonstration made use of FLIGHTLAB, an industry standard software typically used for engineering and piloted evaluations. It is a modular simulator with user-selectable component models of varying fidelity, which the analyst selects based on the needs and constraints of the intended use. Figure C.1 shows a screenshot of the graphical user interface used to select components and define the properties of the helicopter model which is later used with the analysis component of Flightlab. The demonstration in this work utilized the built-in model of the UH-60A helicopter, since this particular model has been validated and contains all necessary components. If a different helicopter model is needed, the appropriate modifications to the model structure would have to be made using the editor shown in Figure C.1, along with modifications to the associated property tables. In the present investigation, LOC events are of particular interest, and specifically the detection of rollover during taxi. In this operating condition, pilots use the lateral cyclic and tail rotor pedal to execute turns with the helicopter, taking care to coordinate these inputs so as to avoid excessive moments and resultant rolling motion. For this type of conditions it is beneficial to have a capability of detecting collisions with the ground or otherwise exceeding the physical ability of the vehicle to sustain safe flight. Therfore a modification to the standard model was made by adding 4 auxiliary contact points, defined as separate landing gear components (shown in Figure C.2), in order to prevent the model from tipping over and crossing the ground place during simulation. The compression of any of these landing gear components is used as a means to detect inappropriate contact with the ground, though any parameters simulated by the complete model could be used in determining the presence

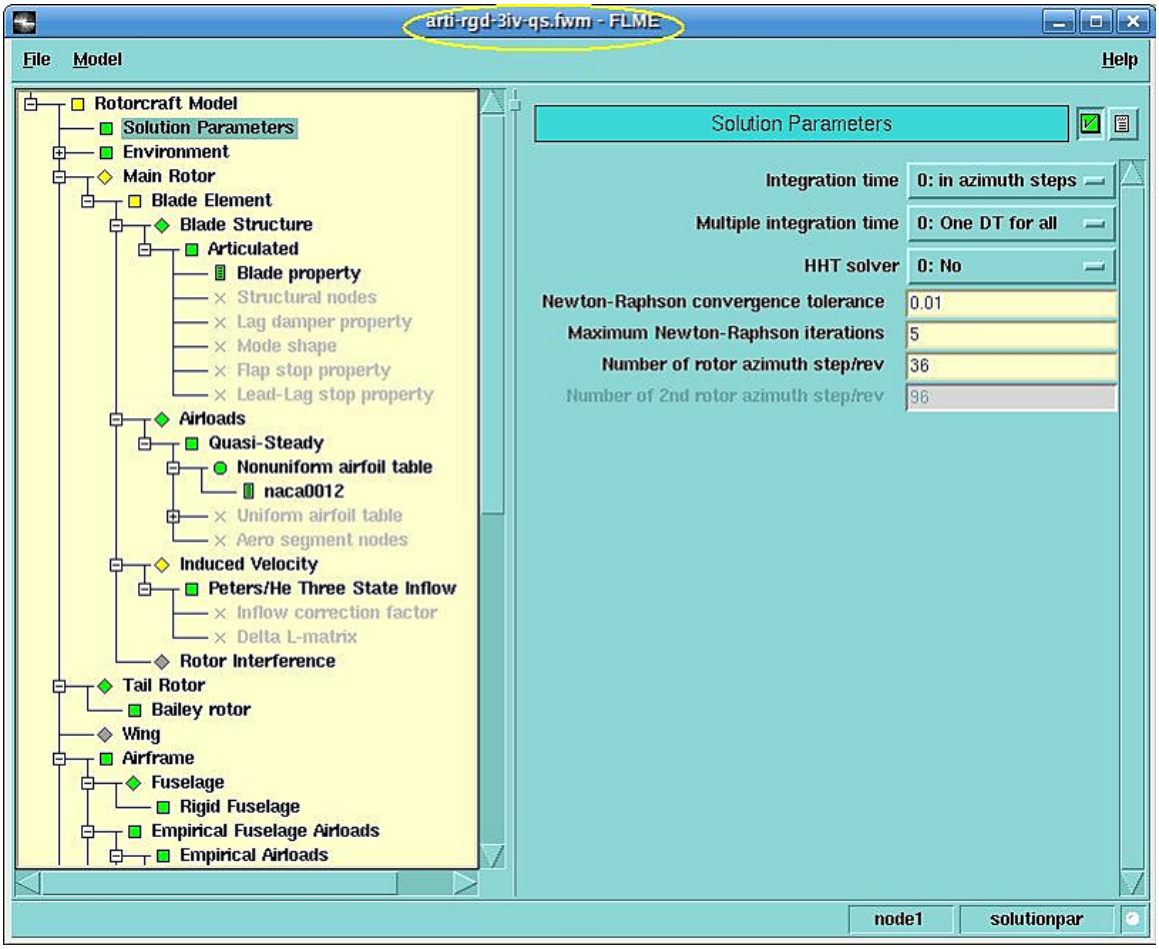


Figure C.1: Flightlab model editor

of an undesirable flight state.

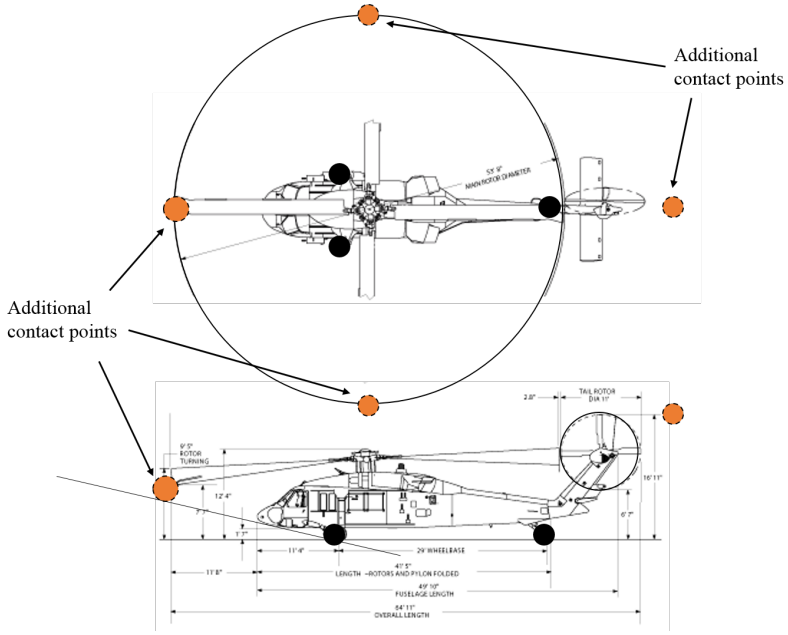


Figure C.2: Added contact points

After a model is defined, the analysis proceeds using the Xanalysis component of Flightlab. After the model is loaded and initialized in Xanalysis, the user can choose the type of analysis to be run. There are many choices, including steady and unsteady simulations in the time domain, frequency response analysis and built in handling quality evaluations. The user can select any of the parameters contained in the model as outputs of the simulation, which can be displayed or saved in the proprietary .TAB format for further processing. Data and simulator configuration can also be loaded from exiting .TAB files. Figure C.3 shows a sample of the outputs that are available, as displayed by the built-in Xanalysis graphing utility.

Both the model editor and the analysis components have a graphical interface, but the software also has a means of scripted interaction with the user through the SCOPE interface in Xanalysis. Scope is a proprietary language, similar in syntax to MATLAB®, which enables the user to generate scripts and automate analyses for which commands exist. The documentation provided with FLIGHTLAB and the help files should be consulted for a

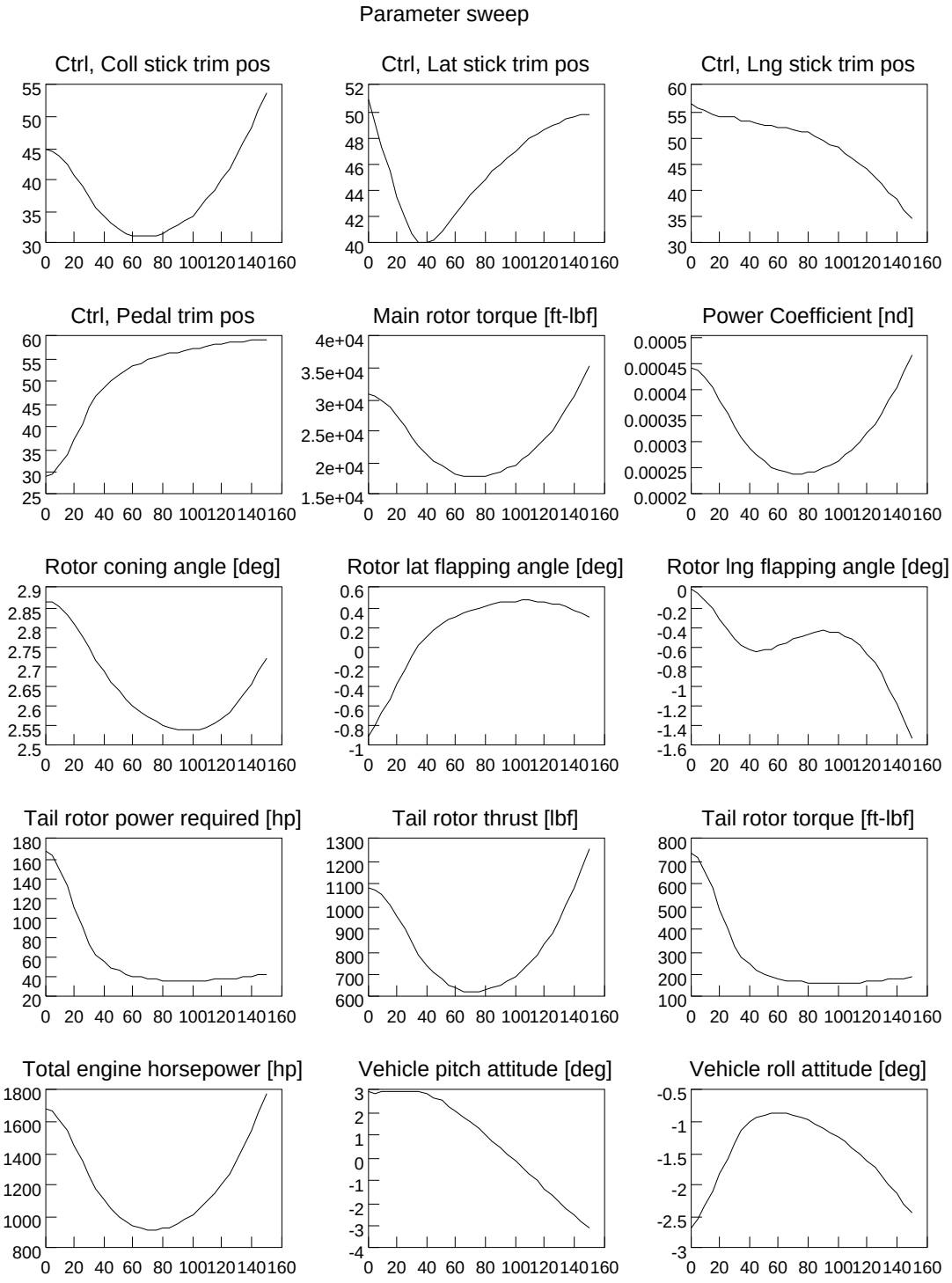


Figure C.3: FLIGHTLAB output

detailed explanation of the various commands and their behavior. This work made use of these facilities to automate the analysis and execute many evaluations in batch format, with the set-up and associated source code described next.

C.2 Flightlab set-up

The simulation set-up was hosted on a Linux computed along with a MATLAB instance which was used to operate the simulation and gather the results. Several files are required to operate the simulation given a range of inputs automatically. The SCOPE environment can execute its own scripts, an example of which is provided in a later subsection. This template is used by the main MATLAB script and modified according to the input for the specific run and the selection of starting condition and vehicle configuration.

C.2.1 MATLAB files

The primary file used to operate the simulation is the mainFL.m file, which calls additional routines that modify various FLIGHTLAB files, and eventually runs a SCOPE script that executes the analysis. The SCOPE script saves the result from each analysis run in a separate file, which can then be processed further. Below are the input files used to simulate the ground maneuvering demonstration.

`mainFL.m:`

```
% Main function to run the analysis using FLIGHTLAB
% Intended to run on Zephram (Linux)

clc
close all
clear
tic
```

```
8
9  %set time of step and integration duration
10 simin.t_final = 10;
11 simin.t_segment = 2;%<<---------- must be less or equal to
       simin.t_final
12 %simin.t_segment can be set to be some smaller interval, as
       long as
13 %it is contained an whole numberr of times in simin.t_final
14
15 simin.t_step_start = 0.5; %the start and stop define the
       step rise time
16 simin.t_step_stop = 0.6;
17
18 % simin.run_folder = 'test_nov11';
19
20 %% determine whether to run ground or free-flight case
21 runtype = 'ground'; % 'trim' 'ground'
22
23 switch runtype
24     case 'ground'
25         %something about modifying the ground run template
               file, which will
26         %be then modified below to be the flightlab_runfile
27         % almost exactly as runfile_template, but now
               groundrun
28         simin.run_folder = 'ground_run_10kt_-10
               coll_extended_test';
```

```
29          unix(['mkdir ~/output_files/' simin.run_folder])
30
31          system_trimfile = '~/input_files/system_ground_10kt_
                -10coll.tab';
32
33          fin = fopen('~/input_files/runfile_template.exc');%
                linux only
34          fout =fopen('~/input_files/ground_run_template.exc',
                'w'); %opened with write permission
35          modify_ground_run_template(fin, fout,
                system_trimfile);
36          fclose(fin); fclose(fout);
37      case 'trim'
38          %% make necessary modification to conditions input
                file
39          %this is for free-flight
40          if simin.t_segment==simin.t_final
41              simin.run_folder = 'parametric_run_continuous';
42          else
43              simin.run_folder = 'parametric_run_multi_segment
                    ';
44          end
45          unix(['mkdir ~/output_files/' simin.run_folder])
46
47          simin.conditions.windaz_val = 0; %knots
48          simin.conditions.windv_val = 0;
49          simin.conditions.windh_val = 0;
```

```
50          simin.conditions.gh_val = 0; %horizontal flight path
                angle (+right)
51          simin.conditions.gv_val = 0; % verticla flight path
               angle (+up)
52          simin.conditions.X_val = 0;
53          simin.conditions.Y_val = 0;
54          simin.conditions.Z_val = -1000; %in feet
55          simin.conditions.veq_val = 10;  %in knots, but gets
               reported in ft/s
56
57          modify_conditions(simin);
58          % remove old and trim to new flight condition
59          unix('rm ~/input_files/system_trim.tab');
60          unix('(scope -i ~/input_files/trimfile_template.exc)
                exit');
61
62      otherwise
63          warning('Unexpected case.')
64  end
65
66  %define values for step control inputs (handled via odli in
       flightlab)
67  xc = [-10];xa=[-30:2:30];xp = xa;
68
69  % reshape into test matrix
70  [xx,yy] = meshgrid(xa,xp);
71  xa_test = reshape(xx,[length(xx)^2,1]);
```

```
72 xp_test = reshape(yy,[length(yy)^2,1]);
73 test_matrix = [xa_test  xp_test [1:length(xa_test)]'];
74
75 particleID = 1; % immediately change after inputs are
       reshaped
76
77 for i = 1:length(xa_test)
78
79 % for i = 1:length(xp)
80 %     for k = 1:length(xa)
81         for j = 1:floor(simin.t_final/simin.t_segment)
82             %set trim values of controls
83             simin.xb = 8;       %longitudinal control,
                   percent
84             simin.xa =test_matrix(i,1);%xa(i);   %lateral
                   control,        percent
85             simin.xc = -10;  %collective
86             simin.xp = test_matrix(i,2);%xp(i);       %tail
                   rotor pedal
87
88             particleID = i;
89
90             %set additional control components (add within
                   modify_controls.m)
91             xb_in = 0;
92             xa_in = 0;
93             xc_in = 0;
```

```
94              xp_in = 0;
95
96              %---> ensure each control input is treated the
                    same
97              %---> ensure I can map control inputs to control
                     outputs (save test cases
98              %in a file, and all output files, for later post
                    -processing)
99
100
101             %use particle_flag to indicate if it is
                    beginning of run or some
102             %intermediate point
103             segmentID = j; %set the segment (can be 1 or a
                    higher number)
104             modify_controls(simin,segmentID) %music In White
                     Rooms - Booka Shade
105
106             %modify primary .exc file for FLIGHTLAB
107             % open files for reading and writing
108             %                       fin = fopen('~/
                    input_files/Xanalysis_template_particle_xh.
                    exc');%linux only
109             switch runtype %this just changes whether the
                    first load is from trim or from pre-defined
                    state
110              case 'trim'
```

```
111                         fin = fopen('~/input_files/
                               runfile_template.exc');%linux only
112                     case  'ground'
113                         fin = fopen('~/input_files/
                               ground_run_template.exc');
114                 end
115                 fout =fopen('FLIGHTLAB_runfile.exc','w'); %
                       opened with write permission
116                 %          fin =fopen('FLIGHTLAB_runfile_test.exc
                       '); %opened with write permission
117
118                 % fin = fopen('test_outputfile.exc')
119
120                 save_name = modify_main_FLfile(simin,segmentID,
                       fin, fout,particleID);
121                 fclose(fin); fclose(fout);
122
123                 %              unix(' timeout -k 1m 30s scope& -i
                        FLIGHTLAB_runfile.exc exit');
124                 unix('(scope -i FLIGHTLAB_runfile.exc exit) &
                       sleep 3; kill $!');
125 %               unix('(scope -i FLIGHTLAB_runfile.exc exit)');
126
127                 %          exec("FLIGHTLAB_runfile.exc",1)
128             end
129 %       end
130 % end
```

```
131
132 end %test matrix
133 simin.test_params = {'xa','xp','particleID'};
134 simin.test_matrix = test_matrix;
135 save(['~/output_files/' simin.run_folder '/simin.mat'],'
    simin')
136
137 runtime = toc;
```

The main file calls additional scripts to modify template files which are used by FLIGHTLAB to execute the analysis. Before a simulation run is executed, the conditions must be set and the control inputs which will be used throughout the simulation are specified using the modify_controls.m and modify_conditions.m scripts, called from the main MATLAB file.

modify_conditions.m:

```
1
2 function modify_conditions(simin,particle_flag,~,~,~)
3 % This script modifies the conditions table that contains
    the initial
4 % conditions used to trim and configure the vehicle. The
    same .tab file is
5 % used during the main execution to ensure the same vehicle
    configuration
6 % is used, though many values are overwritten when the end
    state of the
7 % prior trajectory is loaded.
8 %
```

```
9  %this function depends on having the correct structure
       present in the simin
10 %input structure, of the following type:
11 % simin.conditions.windaz_val = 20;
12 % simin.conditions.windv_val = 60;
13 % simin.conditions.windh_val = 50;
14 % simin.conditions.gh_val = 40;
15 % simin.conditions.gv_val = 30;
16 % simin.conditions.X_val = 20;
17 % simin.conditions.Y_val = 2;
18 % simin.conditions.Z_val = -1000;
19 % simin.conditions.veq_val = 10;
20 %uncomment the above rows if testing the function
21
22 fin = fopen('conditions_input_matlab.tab');
23 fout = fopen('conditions_matlab_generated_input.tab','w');
24 % disp(['filename is open ' num2str(fout)])
25 data_now = 0;
26
27 while data_now ~= -1
28     % get row from file
29     data_now = fgets(fin);
30     data_now = strrep(data_now,'windaz_val',num2str(simin.
           conditions.windaz_val));
31     data_now = strrep(data_now,'windv_val',num2str(simin.
           conditions.windv_val));
```

```
32      data_now = strrep(data_now,'windh_val',num2str(simin.
           conditions.windh_val));
33      data_now = strrep(data_now,'gh_val',num2str(simin.
           conditions.gh_val));
34      data_now = strrep(data_now,'gv_val',num2str(simin.
           conditions.gv_val));
35      data_now = strrep(data_now,'X_val',num2str(simin.
           conditions.X_val));
36      data_now = strrep(data_now,'Y_val',num2str(simin.
           conditions.Y_val));
37      data_now = strrep(data_now,'Z_val',num2str(simin.
           conditions.Z_val));
38      data_now = strrep(data_now,'veq_val',num2str(simin.
           conditions.veq_val));
39
40      fprintf(fout,'%s',data_now); %just pass the row to the
           out file
41
42  end
43
44  fclose(fin);
45  fclose(fout);
46  end
```

The control inputs are set at pre-determined levels and passed as step inputs if the simulation is at the first segment, and can be held constant or varied in later segments.

modify_controls.m:

```
function modify_controls(simin,segmentID, ~, ~, ~)
% This script modifies the input table that contains the
    control commands
% that are executed at a specified time. This version
    generates step inputs
% for the first approach, to replicate the original tipover
    modeling
% Input: tfinal - how long to run,simin.t_step - time of
    step application
% xb - longitudinal
% xa - lateral
% xc - collective
% xp - pedal

% if nargin==1
%     disp('only struct is input')
% else
%     return
% end

fin = fopen('example_input_matlab.tab');
fout = fopen('example_matlab_generated_input.tab','w');
% disp(['filename is open ' num2str(fout)])

data_now = 0;
mode = 1; %start by searching for the control sequence
    definition
```

```
23 while data_now ~= -1
24     % get row from file
25     data_now = fgets(fin);
26
27     %if length is 1, it is an "empty" row between two
           segments, variables,
28     %or vectors; depends on incoming file formatting, works
           with input .tab
29     if length(data_now) == 1
30         mode = 1; %back to search mode if we have an empty
               row (not last)
31     end
32
33     if mode ==1
34         fprintf(fout,'%s',data_now); %just pass the row to
               the out file
35
36         % hardcoded variable title row (ok for small number
               of known
37         % variables such as the control spec file
38         if isequal(data_now,sprintf('!T Timevec  XBin  XAin
               XCin  XPin\n'))
39             mode=2; % switch to modification mode
40         end
41
42     elseif mode == 2 %if in search mode, make the
           modifications below
```

```
43          nums = str2num(data_now);
44
45          if segmentID == 1
46              if  nums(1)>simin.t_step_start
47                  nums(2) = simin.xb; %xbin - longitudinal?
48                  nums(3) = simin.xa; % modify the other 3
                        control channels
49                  nums(4) = simin.xc; % modify the other 3
                        control channels
50                  nums(5) = simin.xp; % modify the other 3
                        control channels
51
52                  if nums(1)==0.5
53                      nums(1)=simin.t_step_start;
54                  end
55                  if nums(1)==0.6
56                      nums(1)=simin.t_step_stop;
57                  end
58                  if nums(1)==8
59                      nums(1)=simin.t_segment;
60                  end
61              end
62          else
63              nums(2) = simin.xb; %xbin - longitudinal?
64              nums(3) = simin.xa; % modify the other 3 control
                    channels
```

```
65                 nums(4) = simin.xc; % modify the other 3 control
                       channels
66                 nums(5) = simin.xp; % modify the other 3 control
                       channels
67
68                 if nums(1)==0.5
69                     nums(1)=simin.t_segment*1/3;
70                 end
71                 if nums(1)==0.6
72                     nums(1)=simin.t_segment*2/3;
73                 end
74                 if nums(1)==8
75                     nums(1)=simin.t_segment;
76                 end
77             end
78
79             data_now = [num2str(nums,' %g') '\n'];
80             fprintf(fout,'%s',sprintf(data_now));
81
82         end
83
84 end
85
86 fclose(fin);
87 fclose(fout);
88 end
```

In the case of ground runs, the analysis proceeded using a saved trim state, which was loaded by replacing the typical trim state with the saved ground trim state.

```
function modify_ground_run_template(fin, fout,
    system_trimfile)
%Created 11/11/2016 by Alek Gavrilovski
%
% this function changes the original template to use the pre
    -trimmed ground
% conditions to start with

data_now = 0;
original_filename = '~/input_files/system_trim.tab';

while data_now ~= -1
    % get row from file
    data_now = fgets(fin);
    %replace original trim file for input with the ground
        run of choice
    data_now = strrep(data_now,original_filename,
        system_trimfile);
    %write each row back to output file
    fprintf(fout,'%s',data_now); %just pass the row to the
        out file
end

%
% fclose(fin);
```

```
% fclose(fout);
end
```

C.2.2 FLIGHTLAB files

The simulation makes use of a file that stores the entire state of the simulator, either as a result of trimming at the beginning of a simulation run or when segmenting the simulation. In the case of trim, this is done to avoid small differences due to the tolerance in the trim algorithm and ensure every run begins at the exact same state. Which file should be used is specified by modification of the template file via the main MATLAB script used to run the analysis. The template FLIGHTLAB file is shown below.

Xanalysis_template.exc:

```
//. This script runs in Xanalysis
//
//.author: Alek Gavrilovski
//.last update: 2016-11-11

//add path to include file locations
path("~/alekmodels");
path("~");

//.Load model (either in Xanalysis or with this command)
   exec("flme/models/articulated/arti-rgd-3iv-qs.def",1);

//.Load input data
    loadtab("~/input_files/example_input.tab");
//  loadtab("example_matlab_generated_input.tab");
```

```
16
17
18
19 // CONFIGURATION
20
21 //the default model and my own modification have: model,
     control, rotor1 and rotor2 components (inertias, cg
     offset, weight for the fuselage; control: analog, digital
      sas, stabilator; rotor1 swashplate phase angle [rad];
     rotor2 failure flag)
22
23   world_model_cpg_configpar_vweight = GW;          // from file
24   world_model_cpg_configpar_fscg    = FSCG;        // from file
25
26
27   //.... etc.
28   exec("xaconfig.exc",1);   //-script set the config
        parameters in model
29
30 // TEST
31
32 //.Set TEST conditions (example: pressure alt, outside temp,
      airspeed)
33 // (all testcond parameters for the gui are in: world_model_
     [component]_cpg_testcond)
34
35 //components: model, airframe, control, propulsion, rotor1
```

```
36
37 // setting up position to coincide with a landing
38     world_model_airframe_cpg_testcond_posxic = 0;
39     world_model_airframe_cpg_testcond_posyic = 0;
40     world_model_airframe_cpg_testcond_poszic = -1000;     //
         make this from file, right now it is meant to test the
          landing
41
42
43     world_model_airframe_cpg_testcond_veq = 10*world_data_f2k
         ; // from file,
44 //    world_model_airframe_cpg_testcond_veq = IAS*
     world_data_f2k; // from file, example unit vonversion
45 // in model, there is ambient pressure/temperature
     disturbance
46     world_model_cpg_testcond_hpres          = HP;
                          // from file
47     world_model_cpg_testcond_tamb           = OAT;
                         // from file (typo in original example
         !!)tran
48    //.... etc.
49    // world_model_airframe_cpg_testcond_veq = Vinf;       //
        VEQ setting flight speed, see how from from file
50     //world_model_airframe_cpg_testcond_wheelhic = Wheelh; //
         from file
51
52 //INSERT TRIM SPECIFICATION FROM PREVIOUS RUN <---
```

```
53 // these values will either be hardcoded or come from file
54
55 world_model_airframe_cpg_testcond_phiic = -0.0321;
56 world_model_airframe_cpg_testcond_thetaic = -0.0142;
57 world_model_control_cpg_testcond_xatrm = 51.47;
58 world_model_control_cpg_testcond_xbtrm = 53.7;
59 world_model_control_cpg_testcond_xctrm = 39.2;
60 world_model_control_cpg_testcond_xptrm = 32.7;
61
62   exec("xatestcond.exc",1);    //-script set the test
       conditions
63
64 //.Trim model at the test condition and with the specified
     configuration
65   exec("xamodeltrim.exc",1);
66
67 // CREATE A SAVEFILE OR OTHER MEANS OF STORING THE TRIM
     STATE (EVERYTHING INCLUDED)
68         pushg // I use push because I want to come back to
              the group where I have been working until now
              world_mytest
69         group outputTrim //I create the group with all the
              output variables. Its name is outputTrim
70
71          //If the variable is 0 (no convergence), it is 1 (
               analysis converged)
```

```
72    trimSuccess = !(world_analysis_trimtest_ifailtrim); //
          Variable to see if there has benn convergence
73
74
75   // if (trimSuccess==1)
76           //Collect trim results. Other place where you can
                 gt information
77 //         shptotal(ikias)=
       world_analysis_trimtest_results_trimout(18);
78 //         mrpwr(ikias)    =
       world_analysis_trimtest_results_trimout(34);
79 //         trpwr(ikias)    =
       world_analysis_trimtest_results_trimout(35);
80 //         ploss(ikias)    =
       world_analysis_trimtest_results_trimout(86);
81           //Read the outputs from the rotor that I am
                interested in
82
83           R1CTTRIM = world_model_rotor1_rotor_cpg_xaout_ct;
84           R1CQTRIM = world_model_rotor1_rotor_cpg_xaout_cq;
85           R1FZTRIM = world_model_rotor1_rotor_cpg_xaout_hubfz;
86           R1CPTRIM = world_model_rotor1_rotor_cpg_xaout_cp;
87           R1CMXTRIM = world_model_rotor1_rotor_cpg_xaout_cmx;
88           R1CMYTRIM = world_model_rotor1_rotor_cpg_xaout_cmy;
89           R1VI0TRIM = world_model_rotor1_rotor_cpg_xaout_vi0
                ;//Induced Flow State Uniform
```

```
90          R1VI1CTRIM = world_model_rotor1_rotor_cpg_xaout_vi1c
                ;//Induced Flow State Cosine
91          R1VI1STRIM = world_model_rotor1_rotor_cpg_xaout_vi1s
                ;//Induced Flow State Sine
92          R1WI0TRIM = world_model_rotor1_rotor_cpg_xaout_wi0
                ;//Mean induced flow
93          R1A1STRIM = world_model_rotor1_rotor_cpg_xaout_a1s
                ;//Swashplate Lateral Angle
94          R1B1STRIM = world_model_rotor1_rotor_cpg_xaout_b1s
                ;//Swashplate Longitudinal Angle
95          R1THETA0TRIM =
                world_model_rotor1_rotor_cpg_xaout_theta0;//
                Swashplate Collective Angle
96          R1MUXYTRIM = world_model_rotor1_rotor_cpg_xaout_muxy
                ;//Advanced Ratio
97          R1OMEGA = world_model_rotor1_rotor_cpg_xaout_omega
                ;//Rotor Speed [rad/sec]
98          //Read the outputs from the model
99          SPHTOTAL = world_model_propulsion_cpg_xaout_shptotal
                ;// Total housepower required [hp]
100         IASFWDXKT = world_model_airframe_cpg_xaout_iasfwdxkt
                ;    //knots
101         VCLIMB= world_model_airframe_cpg_xaout_vclimb; //[ft
                /sec]
102         //Read the outputs from the model
103         MORHOTRIM = world_model_cpg_xaout_rho;
104         MOWINDXTRIM = world_model_cpg_xaout_windx;
```

```
105         MOWINDYTRIM = world_model_cpg_xaout_windy;
106         MOWINDZTRIM = world_model_cpg_xaout_windz;
107         MOTOTVWEIGHT= world_model_cpg_xaout_totvweight; //
                lbm
108         // Eventually I can write other varaibles is
109
110   // end // End trimSuccess==1
111
112   // Store the output variables in the file "
          trimAnalysisOutput.dat"
113   save("~/output_files/trimAnalysisOutput.dat");  //save("
          trimAnalysisOutput.dat");
114
115   popg ;// to go back to the group world_mytest
116
117 //.Select inputs (example: xb, xa, xc, xp)
118 // (this function creates a varlist called "@inputs")
119   input([]);          // clear varlist
120   input( world_model_control_data_xb);
121   input( world_model_control_data_xa);
122   input( world_model_control_data_xc);
123   input( world_model_control_data_xp);
124   //.type: 'who(@inputs)' to see what is going on
125
126 //.Select outputs (example: xb, xa, xc, xp)
127 // (this function creates a varlist called "@outputs")
128   output([]) ;
```

```
129   output(world_model_airframe_cpg_xaout_vxb) ;          //
        column 1
130   output(world_model_airframe_cpg_xaout_vyb) ;
131   output(world_model_airframe_cpg_xaout_vzb) ;
132   output(world_model_airframe_cpg_xaout_p) ;
133   output(world_model_airframe_cpg_xaout_q) ;
134   output(world_model_airframe_cpg_xaout_r) ;
135   output(world_model_airframe_cpg_xaout_phi) ;
136   output(world_model_airframe_cpg_xaout_theta) ;        //
        column 8
137   output(world_model_airframe_cpg_xaout_psi) ;
138   output(world_model_airframe_cpg_xaout_posxi) ;
139   output(world_model_airframe_cpg_xaout_posyi) ;
140   output(world_model_airframe_cpg_xaout_poszi) ;//
141
142
143   //.type: 'who(@outputs)' to see what is going on
144
145 //.Generate input profiles (just as an example)
146 // (load external input data )
147 loadtab("~/example_matlab_generated_input.tab");
148
149 // (example: step in xb )
150    dt    = world_data_dt;        // time step
151    time =  [0:dt:max(timevec)]'; //    time =  [0:dt:8]';
152   //output(time);//add time to the outputs
```

```
153     xbvec= odli(time,timevec,xbin);  // generating input
           profile from file data
154     xavec= odli(time,timevec,xain);  // generating input
           profile from file data
155     xcvec= odli(time,timevec,xcin);  // generating input
           profile from file data
156     xpvec= odli(time,timevec,xpin);  // generating input
           profile from file data
157 //    xavec = xbvec*0.0;  xcvec = xbvec*0.0;  xpvec = xbvec
      *0.0;
158 //    xavec = xbvec*0.0;  xcvec = xbvec*0.0;  xpvec = xbvec
      *0.0;
159
160     yout = nrun([xbvec, xavec, xcvec, xpvec]);   // the same
           sequence as when defining inputs!!
161
162     //.Expand outputs
163     vxb   = yout(:,1);
164     vyb   = yout(:,2);
165     vzb   = yout(:,3);
166     p     = yout(:,4);
167     q     = yout(:,5);
168     r     = yout(:,6);
169     phi   = yout(:,7);
170     theta = yout(:,8);
171     psi   = yout(:,9);
172     x     = yout(:,10);
```

```
173     y       = yout(:,11);
174     z       = yout(:,12);
175     //time          = yout(:,13);
176     //example_Xanalysis_transient_response_with_external_data
            .exc
177     //.Plot results (example: pitch attitude 'theta' - 8'th
            parameter in outputs(...) def
178     xplot("allclear,date=0");
179     xplot("xlabel=Time[sec],ylabel=Nose DWN << Pitch Att [deg
            ] >> Nose UP");
180     xplot(time,theta*57.3);
181
182     //.Save results into a ASCII file
183     //save filename specification in row below
184     savetab("~/output_files/example_results_with_savestates.
            tab",vxb,vyb,vzb,p,q,r,phi,theta,psi,x,y,z,time);
185
186     //Save entire state at end of simulation run so next one
            starts
187     //particle state output
188     //savetab("~/input_files/particle_1.tab",x0);
189
190 //  this is needed if the file is run using "scope -i
        FLIGHTLAB_runfile.exc"
191 exit
192
193 //-EOF-
```

REFERENCES

[1] L Iseler and J. De Maio, "An analysis of us civil rotorcraft accidents by cost and injury (1990-1996)," Ames Research Center, Moffet Field, California, Tech. Rep. May, 2002.

[2] International Helicopter Safety Team. (2015). International helicopter safety team home page. accessed 4/13/2015.

[3] National Transportation Safety Board, "NTSB Most Wanted List 2014: Address Unique Characteristics of Helicopter Operations," 2014, http://www.ntsb.gov/safety/mwl2014/01_MWL_HeliOps.pdf, accessed 04/16/2015.

[4] ——, (2015). NTSB 2015 Most Wanted list of transportation safety improvements. http://www.ntsb.gov/safety/mwl/Documents/MWL_2015_brochure.pdf accessed 4/21/2015.

[5] Boeing Commercial Airplanes. (2014). Statistical Summary of Commercial Jet Airplane Accidents. Worldwide Operations: 1959 - 2013. http://www.boeing.com/news/techissues/pdf/statsum.pdf, accessed 08/2/2015.

[6] P. Goodwin, M. E. Thomson, D. Onkal, and A. Avcio, "Aviation Risk Perception : A Comparison," vol. 24, no. 6, 2004.

[7] F. D. Harris and E. F. Kasper, "U.S. Civil Rotorcraft Accidents, 1963 through 1997," no. December, 1998.

[8] U.S. Joint Helicopter Safety Analysis Team, "The Compendium Report : The U . S . JHSAT Baseline of Helicopter Accident Analysis Volume I," Tech. Rep. August, 2011.

[9] CANADIAN JOINT HELICOPTER SAFETY ANALYSIS TEAM, "Canadian JHSAT Report: 2000 accident data analysis," Tech. Rep., 2000.

[10] National Transportation Safety Board, "Review of u.s. civil aviation accidents - review of aircraft accident data 20072009," National Transportation Safety Board, Washington, DC, Tech. Rep., 2011.

[11] European Helicopter Safety Team, "Analysis of 2000-2005 european helicopter accidents," European Aviation Safety Agency, Cologne, Germany, Tech. Rep., 2010.

[12] Commercial Aviation Safety Team. (2015). Cast — mission, vision, goals. http://www.cast-safety.org/about_vmg.cfm, accessed 4/17/2015.

[13] Civil Aviation Authority, "Cap 739 flight data monitoring," 2013.

[14] L. Williamson, "Monitoring Flight Operations Using Flight Recorded Data," in *AIAA/AHS/ASEE Aircraft Design, Systems and Operations Conference*, Seattle, WA: AIAA, 1989.

[15] J. M. G. F. Stevens and J Vreeken, "The Potential of Technologies to Mitigate Helicopter Accident Factors: An EHEST Study," National Aerospace Laboratory NLR, Tech. Rep., 2014.

[16] V. Klein and E. A. Morelli, *Aircraft System Identification: Theory and Practice*. American Institute of Aeronautics and Astronautics, 2006, ISBN: 1563478323.

[17] M. Tischler and R. Remple, *Aircraft and Rotorcraft System Identification: Engineering Methods with Flight Test Examples*, ser. AIAA education series. American Institute of Aeronautics and Astronautics, 2012, ISBN: 9781600868207.

[18] R. K. Heffley and M. a. Mnich, "Minimum-complexity helicopter simulation math model," 1988.

[19] Federal Aviation Administration, "TSO-C124b Flight Data Recorder Systems," 2007.

[20] BEA, "Flight Data Recorder Read-Out Technical and Regulatory Aspects," Le Bourget Cedex, France, Tech. Rep., 2005.

[21] Joint Aviation Authorities, "JAR-OPS 3 FLIGHT RECORDER REQUIREMENTS EXPLANATORY NOTES," pp. 1–30, 2001.

[22] N. A. H. Campbell, "The Evolution of Flight Data Analysis," pp. 1–22, 2003.

[23] Federal Aviation Administration, "Advisory Circular AC 120-82 Flight Operational Quality Assurance," 2004.

[24] International Civil Aviation Organization, "Flight Data Analysis Programme Manual," 2013.

[25] Petroleum Helicopters Inc., "LAMP Line Activity Monitoring Program," 2009.

[26] National Defense Center for Energy & Environment. (2014). Aircraft Logging and Event Recording for Training and Safety (ALERTS).

http://www.ndcee.ctc.com/index.php/11-technology-article/ 129-aircraft-logging-and-event-recording-for-training-and-safety-alerts, accessed 4/20/2014.

[27] M. A. Morales, D. J. Haas, B. L. Fuller, and T. Chaves, "Leveraging information to support military flight operations quality assurance," in *American Helicopter Society 62nd Annual Forum*, Phoenix, AZ: American Helicopter Society International, Inc., 2009, p. 2006.

[28] Air Force Safety Center, "Annual report: Fiscal year 2012," Tech. Rep., 2012, pp. 1–23.

[29] U.S. Government Publishing Office. (1989). 14 cfr 91.117 aircraft speeds. Doc. No. 18334, 54 FR 34292, Aug. 18, 1989, as amended by Amdt. 91-219, 55 FR 34708, Aug. 24, 1990; Amdt. 91-227, 56 FR 65657, Dec. 17, 1991; Amdt. 91-233, 58 FR 43554, Aug. 17, 1993, accessed 5/11/2015.

[30] Robinson Helicopter Company, *R44 pilot's operating handbook and faa approved rotorcraft flight manual*, Torrance, CA, 2014.

[31] Civil Aviation Department, "Cad 739 flight data monitoring a guide to implementation," Civil Aviation Department, Hong Kong, China, Tech. Rep. 1, 2009.

[32] Civil Aviation Authority, "Final Report on the Follow-on Activities to the HOMP Trial," 2004.

[33] B. Larder, "Final Report on the Helicopter Operations Monitoring Programme (HOMP) Trial," *CAA Paper*, no. 041, 2002.

[34] International Civil Aviation Organization, *Safety Management Manual (SMM)*, 3rd. 2013, ISBN: 9789292492144.

[35] R. V. Fernandes, "An analysis of the potential benefits to airlines of flight data monitoring programmes," 2002.

[36] Federal Aviation Administration, "Fact Sheet Aviation Safety Information Analysis and Sharing (ASIAS) System," no. December, 2011.

[37] D. A. Van Cleave, *Fusing Aviation Data : A New Approach to Keeping Skies Safer*, 2009.

[38] S. K. Lau, "General Aviation Flight Data Monitoring," pp. 1–14, 2007.

[39] ——, "Flight Data Monitoring : General Aviation Safety Information Analysis and Sharing Research Project," CAPACG, LLC., Tech. Rep. January, 2012.

[40] Partnership to Enhance General Aviation Safety, Accessibility and Sustainability. (2015). Rotorcraft asias. https://www.pegasas.aero/projects.php?p=2, accessed 05/11/2015, Federal Aviation Administration.

[41] U.S. JHSIT, "Health and usage monitoring systems toolkit," 2013.

[42] R. Romero, H. Summers, and J. Cronkhite, "Feasibility study of a rotorcraft health and usage monitoring system (hums): usage and structural life monitoring evaluation," Tech. Rep., 1996.

[43] Civil Aviation Authority, "CAP 753: Helicopter Vibration Health Monitoring (VHM)," Tech. Rep., 2012, p. 34.

[44] L Miller, B McQuiston, J Frenster, and D Wohler, "Rotorcraft health and usage monitoring systems-a literature survey," 1991.

[45] I. Delgado, P. Dempsey, and D. Simon, "A survey of current rotorcraft propulsion health monitoring technologies," no. January, 2012.

[46] P. J. Dempsey, D. G. Lewicki, and D. D. Le, "Investigation of current methods to identify helicopter gear health," ... *Conference, 2007 IEEE*, no. July, 2007.

[47] E. Bechhoefer and a. P. F. Bernhard, "A generalized process for optimal threshold setting in HUMS," *IEEE Aerospace Conference Proceedings*, no. Ci, 2007.

[48] D. He, S. Wu, and E. Bechhoefer, "A regime recognition algorithm for helicopter usage monitoring," in *Aerospace Technologies Advancements*, January, T. T. Arif, Ed., InTech, 2010, ISBN: 9789537619961.

[49] U.S. JHSIT, "Helicopter flight data monitoring toolkit," pp. 1–22, 2011.

[50] Federal Aviation Administration. (2012). Helicopter flying handbook. accessed 5/2/2015.

[51] EUROCONTROL. (). Loss of control. http://www.skybrary.aero/ /index.php/Loss_of_Control, accessed 5/21/2015.

[52] J Doerflinger, G. Bruniaux, P. Pezzatini, M. Greiller, and J. Marcellet, "Small Helicopter Operational Monitoring Programme (HOMP) Trial," European Aviation Safety Agency, Tech. Rep., 2010.

[53] Global Helicopter Flight Data Monitoring Steering Group. (2014). Hfdm vendor equipment and services. http://www.hfdm.org/LinkClick.aspx?fileticket=QYqK2qaccessed 5/20/2015.

[54] T. Osbourne. (2014). Airbus helicopters to make black boxes standard. http://aviationweek.com/commercial-aviation/airbus-helicopters-make-black-boxes-standard, accessed 5/20/2015, Aviation Week.

[55] Aviation Information Network, *North Sea Safety Effort Sparks Cooperation among Operators , Manufacturers*, 2015.

[56] M. Schwabacher, N. Oza, and B. Matthews, "Unsupervised anomaly detection for liquid-fueled rocket propulsion health monitoring," *Journal of Aerospace Computing, Information, and Communication*, vol. 6, no. 7, pp. 464–482, 2009.

[57] Aviation Accident Investigation Branch, "Aerospatiale AS332L Super Puma , G-TIGT , 4 January 1996," Tech. Rep. January, 1996.

[58] U. Fayyad, G Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," *AI magazine*, pp. 37–54, 1996.

[59] V. Chandola, A Banerjee, and V Kumar, "Anomaly detection: a survey," *ACM Computing Surveys (CSUR)*, no. September, pp. 1–72, 2009.

[60] C. Bishop, *Pattern recognition and machine learning*. 2006, ISBN: 9780387310732.

[61] E. Keogh and S. Kasetty, "On the need for time series data mining benchmarks: a survey and empirical demonstration," *Data Mining and knowledge discovery*, pp. 349–371, 2003.

[62] O. Maimon and L Rokach, *Data mining and knowledge discovery handbook*. 2005, ISBN: 9780387098227.

[63] T. Hastie, R Tibshirani, and J Friedman, *The elements of statistical learning*. 2009.

[64] I. H. Witten and E. Frank, *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005, ISBN: 978-0-12-374856-0.

[65] B. G. Amidan and T. A. Ferryman, "APMS SVD methodology and implementation," Pacific Northwest National Lab., Richland, WA (US), Tech. Rep., 2000.

[66] I. B. Mugtussidis, "Flight data processing techniques to identify unusual events," PhD thesis, 2000.

[67] D. L. Iverson, "Inductive system health monitoring with statistical metrics," *Proceedings of the 4th JANNAF Modeling & Simulation ...*, 2005.

[68] S. Budalakoti, "Anomaly detection and diagnosis algorithms for discrete symbol sequences with applications to airline safety," *..., Part C: Applications and ...*, 2009.

[69] S. Das, B. L. Matthews, A. N. Srivastava, and N. C. Oza, "Multiple kernel learning for heterogeneous anomaly detection: algorithm and aviation safety case study," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2010, pp. 47–56.

[70] S. Das, L Li, A Srivastava, and R. Hansman, "Comparison of algorithms for anomaly detection in flight recorder data of airline operations," *12th AIAA Aviation Technology ...*, no. September, 2012.

[71] S. Das, S Sarkar, and A Ray, "Anomaly detection in flight recorder data: a dynamic data-driven approach," *American Control*, 2013.

[72] L. Li, M. Gariel, R. J. Hansman, and R. Palacios, "Anomaly detection in onboard-recorded flight data using cluster analysis," *2011 IEEE/AIAA 30th Digital Avionics Systems Conference*, 4A4–1–4A4–11, 2011.

[73] E. Smart, D. Brown, and J. Denman, "A Two-Phase Method of Detecting Abnormalities in Aircraft Flight Data and Ranking Their Impact on Individual Flights," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1253–1265, 2012.

[74] L Li, "Anomaly detection in airline routine operations using flight data recorder data," no. June, 2013.

[75] T Warren Liao, "Clustering of time series data: a survey," *Pattern Recognition*, vol. 38, no. 11, pp. 1857–1874, 2005.

[76] M. Witczak, *Modelling and estimation strategies for fault diagnosis of non-linear systems: From analytical to soft computing approaches*. 2007, vol. 354, pp. 1–208, ISBN: 3540711147.

[77] R. Isermann, *Fault-diagnosis applications: model-based condition monitoring: actuators, drives, machinery, plants, sensors, and fault-tolerant systems*. Springer Science & Business Media, 2011, ISBN: 9783642127663.

[78] M. S. Whalley and M. Achache, "Joint U.S. / France investigation of helicopter flight envelope limit cueing," in *52nd American Helicopter Society Annual Forum*, 1996.

[79] P. Menon, V. Iragavarapu, and M. S. Whalley, "Estimation of Rotorcraft Limit Envelopes using Neural Networks," *Journal of the American Helicopter Society*, p. 1423,

[80] M. Pilgrim, *Intro to hfdm*, `http://www.ihst.org/Portals/54/presentations/08_Intro_to_HFDM_Mike_Pilgrim.pptx`, accessed 04/13/2015, 2014.

[81] S. Burgess, "The Reality of Aeronautical Knowledge : The Analysis of Accident Reports Against What Aircrews are Supposed to Know," Joint Helicopter Measurement and Data Analysis Team, International Helicopter Safety Team, Tech. Rep., 2012.

[82] A. H. Rao and K. Marais, "Identifying High-Risk Occurrence Chains in Helicopter Operations from Accident Data," in *15th AIAA Aviation Technology, Integration, and Operations Conference*, Dallas, TX: American Institute of Aeronautics and Astronautics, 2015.

[83] U.S. Joint Helicopter Safety Analysis Team, "The Compendium Report : The U . S . JHSAT Baseline of Helicopter Accident Analysis Volume I," Tech. Rep. August, 2011.

[84] A. Payan, A. Gavrilovski, H. Jimenez, and D. N. Mavris, "Review of proactive safety metrics for rotorcraft operations and improvements using model-based parameter synthesis and data fusion," in *AIAA Infotech@ Aerospace*, 2016, p. 2133.

[85] D. Gorinevsky, B. Matthews, and R. Martin, "Aircraft anomaly detection using performance models trained on fleet data," *Conference on Intelligent Data Understanding*, pp. 17–23, 2012.

[86] M. B. Tischler, J. W. Fletcher, V. L. Diekmann, R. A. Williams, and R. W. Cason, "Demonstration of frequency-sweep testing technique using a Bell 214-ST helicopter," 1987.

[87] P. Hamel, "Rotorcraft system identification," *AGARD Lecture Series LS*, no. October, 1991.

[88] M. Gevers, "A Personal View of the Development of System Identification," *IEEE Controll Systems*, vol. 26, no. 6, pp. 93–105, 2006.

[89] E. Morelli and J. Cooper, "Frequency-Domain Method for Automated Simulation Updates based on Flight Data," no. January, pp. 1–26, 2014.

[90] J. Sembiring, L. Drees, and F. Holzapfel, "Extracting unmeasured parameters based on quick access recorder data using parameter-estimation method," pp. 1–9, 2013.

[91] J. G. Leishman, *Principles of Helicopter Aerodynamics with CD Extra*. Cambridge university press, 2006.

[92] R. W. Prouty, *Helicopter performance, stability, and control*. 1995.

[93] W. Johnson, *Helicopter theory*. Courier Corporation, 2012.

[94] G. D. Padfield, *Helicopter flight dynamics*. John Wiley & Sons, 2008.

[95] W. J. Hanley and G. DeVore, "An evaluation of the effects of altitude on the height velocity diagram of a single engine helicopter," Federal Aviation Agency Aircraft Development Service, Washington, DC, Tech. Rep., 1964.

[96] ——, "An evaluation of the height velocity diagram of a lightweight, low rotor intertia, single engine helicopter," Federal Aviation Agency Aircraft Development Service, Washington, DC, Tech. Rep., 1965.

[97] W. J. Hanley, G. DeVore, and S. Martin, "An Evaluation Of The Height Velocity Diagram Of A Heavyweight, High Rotor Inertia, Single Engine Helicopter," 1966.

[98] W. J. Hanley and G. DeVore, "An Analysis of the Helicopter Height Velocity Diagram Including a Practical Method for its Determination," Federal Aviation Administration Aircraft Development Service, Atlantic City, NJ, Tech. Rep., 1968.

[99] Y. Okuno, K. Kawachi, A. Azuma, and S. Saito, *Analytical prediction of height-velocity diagram of a helicopter using optimal control theory*, 1991.

[100] Y. Okuno and K. Kawachi, "Optimal control of helicopters following power failure," *Journal of Guidance, Control, and Dynamics*, vol. 17, no. 1, pp. 181–186, 1994.

[101] R. Deresz, "Flight simulation and validation of height-velocity characteristics of an attack helicopter," The Boeing Company, The American Helicopter Society, 2013.

[102] C. Bottasso, "Trajectory optimization of rotorcraft including pilot models, with applications to ADS-33 MTEs, Cat-A procedures and engine off landings," *American Helicopter . . .*, pp. 1–5, 2009.

[103] C. M. Belcastro, L. Groff, R. L. Newman, J. V. Foster, D. a. Crider, D. H. Klyde, and a. McCall Huston, "Preliminary Analysis of Aircraft Loss of Control Accidents : Worst Case Precursor Combinations and Temporal Sequencing," *AIAA SciTech, 13-17 January 2014, National Harbour, MD, USA*, no. AIAA 2014-0612, pp. 1–32, 2014.

[104] J. E. Wilborn and J. V. Foster, "Defining Commercial Transport Loss-of-Control : A Quantitative Approach," *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, no. August, pp. 1–11, 2004.

[105] R. G. Fox, "Lateral Rollover Protection Concepts," Applied Technology Laboratory, US AVRADCOM, Tech. Rep., 1980.

[106] J. Moon and J. V. R. Prasad, "Minimum-time approach to obstacle avoidance constrained by envelope protection for autonomous uavs," *Mechatronics*, vol. 21, no. 5, pp. 861–875, 2011.

[107] S. Unnikrishnan, "Adaptive envelope protection methods for aircraft," PhD thesis, Georgia Institute of Technology, 2006, 187–187 p. ISBN: 9780542862281.

[108] I. Yavrucuk, S. Unnikrishnan, and J. Prasad, "Envelope protection for autonomous unmanned aerial vehicles," *Journal of Guidance, Control, and Dynamics*, vol. 32, no. 1, pp. 248–261, 2009.

[109] M. J. Roemer, C. S. Byington, G. J. Kacprzynski, and G. J. Vachtsevanos, "An Overview of Selected Prognostic Technologies with Application to Engine Health Management," *ASME Turbo Expo 2006: Power for Land, Sea, and Air (GT2006)*, pp. 1–9, 2006.

[110] B.-C. Chen and H. Peng, "A real-time rollover threat index for sports utility vehicles," *Proceedings of the 1999 American Control Conference (Cat. No. 99CH36251)*, vol. 2, no. June, pp. 1233–1237, 1999.

[111] R. Eger and U. Kiencke, "Modeling of rollover sequences," *Control Engineering Practice*, vol. 11, no. 2, pp. 209–216, 2003.

[112] M. E. Dreier, *Introduction to helicopter and tiltrotor simulation*. AIAA (American Institute of Aeronautics and Astronautics), 2007.

[113] P. D. Talbot, B. E. Tinling, W. A. Decker, and R. T. N. Chen, "A mathematical model of a single main rotor helicopter for piloted simulation," no. September, 1982.

[114] A. R. S. Bramwell, G. Done, and D. Balmford, *Bramwell's Helicopter Dynamics*, Second. Woburn, MA: Butterworth-Heinemann, 2001, ISBN: 0 7506 5075 3.

[115] R. T. N. Chen, "A survey of nonuniform inflow models for rotorcraft flight dynamics and control applications," *Vertica*, vol. 14, pp. 147–184, 1990.

[116] C Chen, J. Prasad, and P. Basset, "A simplified inflow model of a helicopter rotor in vertical descent," in *American Helicopter Society 60th Annual Forum, Baltimore, Maryland*, vol. 7, 2004, pp. –10.

[117] J. Wolkovitch, "Analytical Prediction of Vortex-Ring Boundaries for Helicopters in Steep Descents," *Journal of the American Helicopter Society*, vol. 17, no. 3, p. 13, 1972.

[118] D. A. Peters and S.-Y. Chen, "Momentum theory, dynamic inflow, and the vortex-ring state," *Journal of the American Helicopter Society*, vol. 27, no. 3, pp. 18–24, 1982.

[119] B Dang-vu, "Vortex ring state protection flight control law," in *39th European Rotorcraft Forum, Moscow, Russia*, Moscow, 2013.

[120] W. Johnson, "Model for Vortex Ring State Influence on Rotorcraft Flight Dynamics," in *4th Decennial Specialist's Conference on Aeromechanics*, San Francisco, CA: American Helicopter Society International, Inc., 2004.

[121] T. A. Egolf and A. J. Landgrebe, "Helicopter rotor wake geometry and its influence in forward flight. Volume 1: Generalized wake geometry and wake effect on rotor airloads and performance," United Technologies Research Center, East Hartford, CT, Tech. Rep., 1983.

[122] C. He, C. S. Lee, and W. Chen, "Technical Note: Finite State Induced Flow Model in Vortex Ring State," *Journal of the American Helicopter Society*, vol. 45, no. 4, p. 318, 2000.

[123] C. Chen and J. V. R. Prasad, "Simplified rotor inflow model for descent flight," *Journal of Aircraft*, vol. 44, no. 3, pp. 936–944, 2007.

[124] W. R. Sturgeon and J. D. Phillips, "A Mathematical Model of the CH-53 Helicopter," Ames Research Center, Moffet Field, California, Tech. Rep., 1980.

[125] C. Roos, J. M. Biannic, S. Tarbouriech, C. Prieur, and M. Jeanneau, "On-ground aircraft control design using a parameter-varying anti-windup approach," *Aerospace Science and Technology*, vol. 14, no. 7, pp. 459–471, 2010.

[126] P. Evans, M. G. Perhinschi, and S. Mullins, "Modeling and Simulation of a Tricycle Landing Gear at Normal and Abnormal Conditions," no. August, pp. 1–20, 2010.

[127] A. G. Barnes and T. J. Yager, "Enhancement of Aircraft Ground Handling Simulation Capability," NATO, Tech. Rep., 1998.

[128] J. Sembiring, L. Hohndorf, and F. Holzapfel, "Bayesian approach implementation on quick access recorder data for estimating parameters and model validation," no. June, 2014.

[129] J. Blackwell and R. Feik, "A Mathematical Model of the On-Deck Helicopter/Ship Dynamic Interface," Aeronautical Research Labs, Melbourne, Australia, Tech. Rep. No. ARL-AERO-TM-405. 1988.

[130] R. Langois and A. Scribner, "Probabilistic Dynamic Interface Analysis Using Monte Carlo Simulation Techniques,"

[131] D. R. Linn and R. G. Langlois, "Development and Experimental Validation of a Shipboard Helicopter On-Deck Maneuvering Simulation," *Journal of Aircraft*, vol. 43, no. 4, pp. 895–906, 2006.

[132] M. J. Leveille, "Development of a Spacial Dynamic Handling and Securing Model for Shipboard Helicopters," PhD thesis, Carleton University, 2013.

[133] R. F. Smiley and W. B. Horne, "Mechanical Properties of Pneumatic Tires with Special Reference to Modern Aircraft Tires," 1958.

[134] H. Pacejka, *Tire and vehicle dynamics*. Elsevier, 2005.

[135] H. B. Pacejka and E. Bakker, "The magic formula tyre model," *Vehicle system dynamics*, vol. 21, no. S1, pp. 1–18, 1992.

[136] J. Svendenius and B. Wittenmark, "Brush tire model with increased flexibility," no. 1,

[137] C. Canudas-de Wit, P. Tsiotras, E. Velenis, M. Basset, and G. Gissinger, "Dynamic Friction Models for Road/Tire Longitudinal Interaction," *Vehicle System Dynamics*, vol. 39, no. 3, pp. 189–226, 2003.

[138] United States Army Aviation and Missle Command Aviation Engineering Directorate, "Aeronautical design standard performance specification handling qualities requirements for military rotorcraft," Tech. Rep., 2000.

[139] C. L. Blanken, R. H. Hoh, D. G. Mitchell, and D. L. Key, "Test guide for ads-33e-prf," DTIC Document, Tech. Rep., 2008.

[140] C. L. C. Bottasso, A. Croce, D. Leonello, and L. Riviello, "Optimization of Critical Trajectories for Rotorcraft Vehicles," *Journal of the American ...*, vol. 50, no. February, pp. 165–177, 2005.

[141] M. W. Floros, "DESCENT Analysis for Rotorcraft Survivability with Power Loss," *Security*, 2009.

[142] I. Burdun, "Safety windows: knowledge maps for accident prediction and prevention in multifactor flight situations," *27th Congress of the International Council of ...*, 2010.

[143] A. Ansari, M.-j. Yu, and D. S. Bernstein, "Exploration and mapping of an unknown flight envelope," in *IEEE Conference on Decision and Control*, 2014, ISBN: 9781467360883.

[144] M Orchard and G Vachtsevanos, "A Particle Filtering Approach for On-Line Fault Diagnosis and Failure Prognosis," *Measurement And Control*, vol. 31, no. 3-4, pp. 1–18, 2004.

[145] A. Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing : fiteen years later," no. December, pp. 4–6, 2011.

[146] N. A. Melchior and R. Simmons, "Particle RRT for path planning with uncertainty," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 1617–1624, 2007.

[147] H. J. Pradlwarter and G. I. Schuëller, "Assessment of low probability events of dynamical systems by controlled Monte Carlo simulation," *Probabilistic Engineering Mechanics*, vol. 14, no. 3, pp. 213–227, 1999.

[148] P. Turati, N. Pedroni, and E. Zio, "An Adaptive Simulation Framework for the Exploration of Extreme and Unexpected Events in Dynamic Engineered Systems.," *Risk analysis : an official publication of the Society for Risk Analysis*, 2016.

[149] B. D. Ng, A. Pfeffer, R. Dearden, and B. Ng, "Continuous Time Particle Filtering," *International Joint Conferences on Artificial Intelligence*, 2005.

[150] A. Gavrilovski and D. N. Mavris, "A model-based approach for event definition in support of flight data monitoring," in *41st European Rotorcraft Forum, Munich, Germany*, 2015.

[151] G. J. J. Jeram, "Open platform for limit protection with carefree maneuver applications," PhD thesis, Georgia Institute of Technology, 2004.

[152] R. H. Myers, D. C. Montgomery, and C. M. Anderson-Cook, *Response surface methodology: process and product optimization using designed experiments*. John Wiley & Sons, 2016.

[153] P. Qiu, "Discontinuous regression surfaces fitting," *Annals of Statistics*, vol. 26, no. 6, pp. 2218–2245, 1998.

[154] Y. LeCun, Y. Bengio, G. Hinton, L. Y., B. Y., and H. G., "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015. arXiv: arXiv:1312.6184v5.

[155] L. E. Fowler, "A virtual pilot algorithm for synthetic hums data generation," Masters thesis, Georgia Institute of Technology, 2015.

[156] M. Lone and A. Cooke, "Review of pilot models used in aircraft flight dynamics," *Aerospace Science and Technology*, vol. 34, pp. 55–74, 2014.

[157] E. A. Wan and A. A. Bogdanov, "Model predictive neural control with applications to a 6 DoF helicopter model," *2001 American Control Conference*, vol. 1, no. August, pp. 488–493, 2001.

[158] C. L. Bottasso, G. Maisano, and F. Scorcelletti, "Trajectory Optimization Procedures for Rotorcraft Vehicles, Their Software Implementation, and Applicability to Models of Increasing Complexity," *Journal of the American Helicopter Society*, vol. 55, no. 3, p. 032 010, 2010.

[159] C. L. Bottasso and P. Montinari, "Rotorcraft Flight Envelope Protection by Model Predictive Control," *Journal of the American Helicopter Society*, vol. 60, no. 022005, pp. 1–13, 2015.

[160] B. Gassaway, K. Strope, L. Cicolani, J. Lusardi, C. He, and D. Robinson, "Predictive Capabilities of a UH-60 FLIGHTLAB ® Model with an External Sling Load," 2006.

[161] E. Head, "Undetermined reasons," *Vertical Magazine*, 2016, https://www.verticalmag.com/features/undetermined-reasons/ accessed 12/10/2016.